Robert Krimmer · Melanie Volkamer · David Duenas-Cid · Peter Rønne · Micha Germann (Eds.)

LNCS 13553

Electronic Voting

7th International Joint Conference, E-Vote-ID 2022
Bregenz, Austria, October 4–7, 2022
Proceedings

Springer

Lecture Notes in Computer Science 13553

More information about this series at https://link.springer.com/bookseries/558

Robert Krimmer · Melanie Volkamer ·
David Duenas-Cid · Peter Rønne ·
Micha Germann (Eds.)

Electronic Voting

7th International Joint Conference, E-Vote-ID 2022
Bregenz, Austria, October 4–7, 2022
Proceedings

Springer

Editors
Robert Krimmer
Tallinn University
Tallinn, Estonia

David Duenas-Cid
University of Tartu
Tartu, Estonia

Micha Germann
University of Bath
Bath, UK

Melanie Volkamer
Karlsruhe Institute of Technology
Karlsruhe, Germany

Peter Rønne
University of Luxembourg
Esch-sur-Alzette, Luxembourg

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-031-15910-7 ISBN 978-3-031-15911-4 (eBook)
https://doi.org/10.1007/978-3-031-15911-4

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains papers presented at E-Vote-ID 2022, the Seventh International Joint Conference on Electronic Voting, held during October 4–7, 2022. This was the first in-person conference following the COVID-19 pandemic, and, as such, it was a very special event for the community since we returned to the traditional venue in Bregenz, Austria. The E-Vote-ID conference resulted from merging EVOTE and Vote-ID, and 18 years have now elapsed since the first EVOTE conference in Austria. Since that conference in 2004, over 1500 experts have attended the venue, including scholars, practitioners, authorities, electoral managers, vendors, and PhD students. E-Vote-ID collects the most relevant debates on the development of electronic voting, from aspects relating to security and usability through to practical experiences and applications of voting systems, also including legal, social, or political aspects, amongst others, turning out to be an important global referent on these issues.

Also, this year, the conference consisted of

- Security, Usability, and Technical Issues Track;
- Governance Track;
- Election and Practical Experiences Track;
- PhD Colloquium;
- Poster and Demo Session.

E-Vote-ID 2022 received 34 submissions for consideration in the first two tracks. After the submission deadline, the Programme Committee members of the respective tracks bid for the papers to review: the respective track chairs assigned the papers, with the aim to have each reviewed by three to five Program Committee members using a double-blind review process. After the completion of the reviews, the track chairs led a discussion with the reviewing Programme Committee members regarding (conditional) acceptance or rejection. For a conditional acceptance, a shepherd was assigned to ensure that the reviewers' proposed changes were included and the revised paper could be accepted. Finally, after a joint discussion, the general chairs made the final decisions with the track chairs. As a result, 10 papers were accepted for this volume, representing 29% of the submitted proposals. The selected papers cover a wide range of topics connected with electronic voting, including experiences and revisions of the actual uses of E-voting systems and corresponding processes in elections.

We would like to thank the German Informatics Society (Gesellschaft für Informatik), with its ECOM working group, and KASTEL for their partnership over many years. Further, we would like to thank the Swiss Federal Chancellery and the Regional Government of Vorarlberg for their kind support. E-Vote-ID 2022 was kindly supported through the European Union's Horizon 2020 projects ECEPS (grant agreement 857622) and mGov4EU (grant agreement 959072). Special thanks go to the international Programme Committee members for their hard work in reviewing, discussing, and shepherding papers. They ensured the high quality of these proceedings with their knowledge and experience.

October 2022

Robert Krimmer
Melanie Volkamer
David Duenas-Cid
Peter Roenne
Micha Germann

Organization

General Chairs

Robert Krimmer	University of Tartu, Estonia
Melanie Volkamer	Karlsruhe Institute of Technology, Germany
David Duenas-Cid	Gdansk University of Technology, Poland, and University of Tartu, Estonia

Security, Usability, and Technical Issues Track Chairs

Melanie Volkamer	Karlsruhe Institute of Technology, Germany
Peter Roenne	Université de Lorraine, Loria, CNRS, France

Governance Track Chairs

Robert Krimmer	University of Tartu, Estonia
Micha Germann	University of Bath, UK

Program Committee

Marta Aranyossy	Corvinus University of Budapest, Hungary
Roberto Araujo	Universidade Federal do Pará, Brazil
Jordi Barrat i Esteve	Universitat Rovira i Virgili, Spain
Bernhard Beckert	Karlsruhe Institute of Technology, Germany
Josh Benaloh	Microsoft, USA
Matthew Bernhard	University of Michigan, USA
Enka Blanchard	Université de Lorraine, France
Jurlind Budurushi	IT University of Copenhagen, Denmark
Jeremy Clark	Concordia University, Canada
Cesar A. Collazos	Universidad del Cauca, Colombia
Veronique Cortier	CNRS, Loria, France
Régis Dandoy	Universidad San Francisco de Quito, Ecuador
Staffan Darnolf	International Foundation for Electoral Systems, USA
Constantin Catalin Dragan	University of Surrey, UK
David Duenas-Cid	Gdansk University of Technology, Poland
Helen Eenmaa	University of Tartu, Estonia
Aleksander Essex	University of Western Ontario, Canada
Chelsea Gabel	McMaster University, Canada

Micha Germann	University of Bath, UK
J. Paul Gibson	Institut Mines-Télécom, France
Rosario Giustolisi	IT University of Copenhagen, Denmark
Kristian Gjøsteen	Norwegian University of Science and Technology, Norway
Nicole Goodman	Brock University, Canada
Rajeev Gore	Australian National University, Australia
Ruediger Grimm	University of Koblenz, Germany
Rolf Haenni	Bern University of Applied Sciences, Switzerland
Thomas Haines	Queensland University of Technology, Australia
Thomas Hofer	Objectif Securité, Switzerland
Bart Jacobs	Radboud University, The Netherlands
Wojtek Jamroga	Polish Academy of Sciences, Poland
Norbert Kersting	University of Münster, Germany
Michael Kirsten	Karlsruhe Institute of Technology, Germany
Reto Koenig	Bern University of Applied Sciences, Switzerland
Steve Kremer	Inria, France
Robert Krimmer	University of Tartu, Estonia
Iuliia Krivonosova	Tallin University of Technology, Estonia
Oksana Kulyk	IT University of Copenhagen, Denmark
Ralf Küsters	University of Stuttgart, Germany
Andreas Mayer	Hochschule Heilbronn, Germany
Johannes Mueller	University of Luxembourg, Luxembourg
Magdalena Musial-Karg	Adam Mickiewicz University, Poland
Andras Nemeslaki	Budapest University of Technology and Economics, Hungary
Stephan Neumann	Landesbank Saar, Germany
Hannu Nurmi	University of Turku, Finland
Jon Pammett	Carleton University, Canada
Olivier Pereira	UCLouvain, Belgium
Pascal Reisert	University of Stuttgart, Germany
Karen Renaud	University of Strathclyde, UK
Adria Rodriguez	Scytl Election Technologies, Spain
Peter Roenne	Université de Lorraine, Loria, CNRS, France
Stefan Roseman	Federal Office for Information Security, Germany
David Ruescasn	nVotes, Spain
Mark Ryan	University of Birmingham, UK
Peter Y. A. Ryan	University of Luxembourg, Luxembourg
Giulia Sandri	European School of Political and Social Sciences, France
Peter Sasvari	National University of Public Service, Hungary
Steve Schneider	University of Surrey, UK

Berry Schoenmakers	Eindhoven University of Technology, The Netherlands
Carsten Schuermann	IT University of Copenhagen, Denmark
Ted Selker	University of California, Berkeley, USA
Uwe Serdült	Ritsumeikan University, Japan
Rodney Smith	University of Sydney, Australia
Mihkel Solvak	University of Tartu, Estonia
Philip Stark	University of California, Berkeley, USA
Ewa Syta	Yale University, USA
Vanessa Teague	Thinking Cybersecurity, Australia
Tomasz Truderung	University of Trier, Germany
Siim Trumm	University of Nottingham, UK
Priit Vinkel	e-Governance Academy, Estonia
Melanie Volkamer	Karlsruhe Institute of Technology, Germany
Felix von Nostitz	Université Catholique de Lille, France
Roland Wen	University of New South Wales, Australia
Jan Willemson	Cybernetica, Estonia
Filip Zagorski	Wroclaw University of Technology, Poland
Marie-Laure Zollinger	Université du Luxembourg, Luxembourg

Contents

An Analysis of the Security and Privacy Issues of the Neovote Online Voting System

Enka Blanchard[1,2,3](✉), Antoine Gallais[2], Emmanuel Leblond[4], Djohar Sidhoum-Rahal[5], and Juliette Walter[6]

[1] CNRS, Paris, France
enka.blanchard@cnrs.fr
https://koliaza.com
[2] Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines, UPHF, Valenciennes, France
[3] Centre Internet et Société, CNRS, Paris, France
[4] Scille SAS, Saint-Médard-en-Jalles, France
[5] Observatoire des Mutations Institutionelles et Juridiques, Université de Limoges, Limoges, France
[6] Unite Live, Paris, France

Abstract. This article provides the first security and privacy analysis of the Neovote voting system, which was used for three of the five primaries in the French 2022 presidential election. We show that the demands of transparency, verifiability and security set by French governmental organisations were not met, and propose multiple attacks against the system targeting both the breach of voters' privacy and the manipulation of the tally. We also show how inconsistencies in the verification system allow the publication of erroneous tallies and document how this arrived in practice during one of the primary elections.

Keywords: Cybersecurity · Electronic voting system · Privacy · Case study · Online voting

1 Introduction

Voting is commonly associated with the sovereign expression of a voter's will—an expression which multiple theorems from voting theory have shown not to be necessarily aligned with their true desires [28]. We now understand that an important pre-condition for the expression of a voter's will is the privacy of the ballot, which was not always the case [6,22,27] and is still debated—e.g. for ballot selfies [20]. However, most individuals' voting experiences happen not in national elections but in small settings such as boardrooms or union meetings—which form the main client base of Neovote. This creates new issues when it comes to privacy, as authority figures can often easily exert direct coercive power if the vote goes against their wishes [4]. The coercive power of managers within

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 1–18, 2022.
https://doi.org/10.1007/978-3-031-15911-4_1

a company—and the related potential for vote buying—should not be neglected, as they are central to large-scale voting fraud campaigns such as those in Russia [14,18]. Thus, if we seek to change or enforce norms that guarantee voters' privacy, smaller-scale votes are an immediate target. The potential for coercion also means that such settings require even more stringent guarantees of privacy, such as publishing not the full tally (from which one could infer some information) but only the result [15]. Before the Covid-19 pandemic, most votes of this kind still happened with paper ballots. The switch to remote work has made this impossible in many settings, leading to the development and partial adoption of many e-voting systems. Some focus on security and privacy [26], whereas others are purposefully designed with usability and understandability in mind, discarding entirely the question of coercion [7].

This article focuses on one such family of voting systems, created in 2007 by a company called Neovote. This company was relatively discreet until 2017, at which point they started establishing themselves as a market leader (they now announce handling upwards of 10 000 elections per year). Since then they have been used for internal elections by at least 245 companies and institutions[1]. They have also won[2] at least 21 competitive public markets—half of which come from academic institutions—valued up to 1.28M€ each for a total of more than 6.5M€ (some contracts do not specify amounts). More importantly, they were recently chosen to organise 3 of the 5 main primary elections for the French Presidential election of 2022. Those are the primary for Les Républicains (LR, two rounds in December 2021), the Primaire de l'Ecologie (PE, two rounds in September 2021), and the Primaire Populaire (PP, single round using majority judgement in January 2022).

As actors like Neovote take an increasingly central role in both democratic institutions and private organisations, it stands to reason that they should be scrutinised and audited by independent actors. They have in practice mostly received attention from the press, especially with the recent primaries [29]. This included some criticism, mostly focused on the fact that the votes they organised were happening online [30] and required some private information (phone and credit card information), as well as for the fact that they allowed certain people to vote twice by using multiple phones and credit cards [19]. The problem of making sure that the remote voter's identity is correct is a central one, to which no solution has yet been found and accepted. However, beyond these issues which apply to all e-voting systems, there remains the question of whether Neovote attains its other security objectives—such as guaranteeing the integrity of the tally or the privacy of voters.

The only other academic work on Neovote was performed by researchers and students from Bordeaux University's computer science department and put online on February 18th, 2022 (as this article was being written). As their

[1] Based on the partial information available on Legifrance, the official French government website for legal information https://www.legifrance.gouv.fr/.

[2] This information was gathered using BOAMP, the French *Official Bulletin for Public Market Offers*, https://www.boamp.fr/.

university was using Neovote for their internal elections, they had a unique opportunity to study it. They showed multiple vulnerabilities—including some affecting privacy, such as the use of ESMTPS (Extended Mail Transfer Protocol), which allows man-in-the-middle attacks during password recovery [5]. They criticised the weakness of some of the registration procedures, dependent on information considered "private" but that would be available to many colleagues (and trivially obtainable by many human resources departments), such as full name, place and date of birth, or student/personnel number (shown on some access badges).

Unlike the Bordeaux team, we did not have access to a small-scale vote which we could participate in. Instead, we decided to run a purely observational study on the PP vote where some of us were legitimately registered. We voted as standard citizens without trying to artificially affect the outcome. All the while, we recorded both our actions in screen-capture and what happened in our browsers, keeping a copy of all html files and scripts that were downloaded[3]. One of the authors of this article was initially a whistle-blower who registered for the PE vote and tried to make public some of the information observed during this vote[4]. This author then shared the corresponding files for the verification process, which the rest of us then authenticated.

This article presents three results concerning Neovote's online voting system:

- The voting system does not respect French security and privacy regulations and does not fulfill its own claims, especially when it comes to vote verifiability.
- The lack of end-to-end verification creates an opportunity for both errors and attacks, leading to the temporary publication of an erroneous tally in the Primaire Populaire vote, which we document here.
- The verification process does not give voters any guarantees as to the integrity of the voting process. However, it allows multiple attacks that could deanonymise some or even all voters.

This article starts by looking at the legal and regulatory constraints that apply to the French e-voting ecosystem, and follows with a first set of issues with Neovote concerning transparency and inconsistent claims. We then perform a code comparison between the public Neovote code and the obsolescent and unmaintained asmcrypto.js library, showing that some of the cryptographic primitives used in the first were directly taken from the latter. We follow with the errors we documented in the online tally published on the official website of the PP election. We then look at the verification procedures proposed by Neovote and show how they do not provide any guarantee on the integrity of the procedure, while still allowing an adversary to breach the voters' privacy in multiple configurations. Finally, we look at how the legal system has reacted to

[3] The corresponding video, html and script files are available upon request.

[4] He also warned the company as well as relevant institutions (ANSSI and CNIL) before waiting the regulatory three months, talking to us and warning some journalists.

these issues (with a focus on France) and conclude on general recommendations to ensure the privacy of voters beyond the strict confines of national elections.

2 Legal and Regulatory Constraints in France

France strongly differentiates between state-organised votes (referenda and elections) and other kinds of votes (many of which concern internal representation in private companies, unions, and institutions). This also includes votes for primary elections, which are poorly covered by French law [24]. Besides some restricted experiments on electronic (offline) voting machines, state-organised votes are all paper-based for residents of mainland France [13]. The second kind of vote is thus a place for experimentation, socio-political as well as technological. The Primaire Populaire combined both by having online voting as well as a not-yet-standard evaluative voting system called majority judgement [3].

The main regulations that apply to votes handled by companies like Neovote then come from two sources. The first is the National Commission on Informatics and Liberty (Commission nationale de l'informatique et des libertés, CNIL) who established some guidelines on electronic and online voting, updated for the last time in 2019 [9]. These guidelines define three risk levels dependent on the importance and scale of the vote and the assumed capabilities of adversaries. Each level adds a supplementary set of constraints. We will focus here on five specific ones which are the most relevant to the problems we consider.

- Security objective n° 1-07: Ensure the total separation of the voter's identity and the expression of their vote for the whole processing duration.
- Security objective n° 1-11: Ensure that the opening of the ballot box and the tallying of its contents can be verified a posteriori.
- Security objective n° 2-06: Use an information system that puts into place the physical and computational security measures recommended by the publishers of the information system and by the National Cybersecurity Agency of France (ANSSI).
- Security objective n° 2-07: Ensure the transparency of the ballot box for all voters.
- Security objective n° 3-02: Allow all voters to check the transparency of the ballot box using third-party tools.

Neovote states that they are homologated (i.e. have received government certification) to organise votes at the highest risk level. This means that they also have to follow the ANSSI regulations, especially the Selection Guide for Cryptographic Algorithms [1]. This guide restates two common cryptographic statements—derived from Kerckoffs' principle—by discouraging from creating new protocols (2.2.6), but most importantly not trying to reimplement standard tools (2.2.5): “This is why it is imperative to only use libraries which have been tested and which benefit from regular maintenance on their security protocols for any use of cryptographic mechanisms.” [1].

If we want to restate in more standard academic language the properties required by the CNIL regulations, we can assimilate n° 1-07 to ensuring the privacy of the vote [2]. Properties n° 2-07 and n° 3-02 correspond to a mix between *cast as intended* and *recorded as cast* (as it implies that voters should be able to check that their vote was indeed recorded, although this is not explicitly stated) [2,32]. Property n° 1–11 would then be closest to *tallied as recorded.*

One curious detail that we discovered during this investigation is that the constraint of code transparency is often left unsaid in academic verifiable voting, especially when proposing new systems—except for a few recent exceptions [16,17]. Most E2E-verifiable voting systems are open-source [2,10], and the availability of both the code and the protocol are often considered a given. However, it is at best rarely explicitly stated that vote verifiability depends on the openness of the source-code (or at least the openness on the protocols used). However, one can wonder how a system could be verifiable if it integrates some black-box components—or at least, if those components do not have strong constraints on all inputs and outputs. This problem and its implications for the online voting technology ecosystem will be explored in more details in Sect. 7. Before we get to this, we need to analyse how the different claims made by Neovote relate to these regulations and how the voting system developed fails to comply with them on multiple fronts.

3 Claims and Transparency

We can start our analysis of Neovote by its public-facing website. It features a number of claims which are either hard to make sense of or inconsistent, as seen on the two following examples.

- They claim to be homologated by many top-level French institutions (Senate, National Assembly, Ministry of the Interior, Council of State), none of which generally practice homologation. When pressed to document or explain these claims by colleagues, they gave no response [5].
- Some of their technical vocabulary is far from standard and is not defined (geometric models for ballot boxes, random ballot boxes). They also reject certain standard tools and claim not to use any kind of database, as well as no "mélangeurs" (mixers or mix-nets). They also indicate being deployed on SecNumCloud servers (a French certification for cloud security) while also saying they do not use any cloud resources.

One central claim they make is that they develop all their code internally, including a modified Debian[5] OS and a full cryptographic toolset, which goes against the ANSSI guidelines. However, as the next section will show, it also is not correct, and some of their code is taken from external libraries. However,

[5] One peculiar element regarding this is that some of the voting archives available from their servers are encrypted with utf8 passwords, and others in iso-8859-1, generally only used on Windows.

before proving this element, we must address a first issue with the Neovote system: it offers no information on its inner workings. This means that we had no source code, no documentation, no white paper or technical paper on the protocol they use, or even any information on who designed the system or implemented it (besides that some of them were trained in a French engineering school). Neovote claims that this lack of transparency—for example, on how they deal with cyberattacks—is due to legal obligations as they handle "top-secret" information [19] even as their CEO claims that e-voting is legitimate for national elections "as long as the solution is ethical and transparent" [29]. This limits the analyses and security tests that can be made ethically, and as such the following sections only feature what could be gleaned from client-side information.

A second point that compounds with the first issue is that multiple elements make this analysis harder. First, the available code—two Javascript files called by the html of the voting interface—was partially obfuscated. Second, despite our team using multiple browsers and operating systems, no-one managed to directly download the .har files of the information exchanged by the browsers during the experiment (the contents then had to be manually copied and saved in text files). This does not directly indicate malicious intent against external observations and was initially thought to be a bug, but still merits a mention. Finally, both the voting website (with a neovote.fr domain name) and the Primaire Populaire website refused to interact with the wayback machine (archive.org), limiting the ability to get a "neutral" external copy of our observations.

4 Code Re-use from Asmcrypto

As stated previously and despite their claims to having all their code developed internally, we found multiple examples of code re-use by Neovote. This is not by itself a problem and we are entirely in favour of code re-use, especially in security contexts, when the code is open-source, maintained and frequently audited. The code we initially downloaded from Neovote was not fully minified but partially obfuscated, with variable names and function names replaced by arbitrary strings (with some structure in the name format). This is apparently done each time the scripts are downloaded from the server. However, certain function names (such as *aes_init*) were not fully replaced, probably to address dependencies, and neither were strings shown to the client. This is what allowed the initial discovery of the asmcrypto.js library, available on Github under an MIT license. From then, we could find a certain number of functions copied and transpiled from one project to the other (from typescript to javascript). We'll look at two main examples and why they are relevant to our considerations.

4.1 General Copy

The first example we look at is linked to the AES encryption of the ballots. Starting with aes.ts of the asmcrypto.js library[6], we can find on lines 55–96 the

[6] https://github.com/asmcrypto/asmcrypto.js/blob/master/src/aes/aes.ts.

```
AES_Encrypt_process
  (data: Uint8Array):Uint8Array
  {if (!is_bytes(data)) throw new
    TypeError("data isn't of
      expected type");
    let asm = this.asm;
    let heap = this.heap;
    let amode = AES_asm.ENC[this.
      mode];
    let hpos = AES_asm.HEAP_DATA;
    let pos = this.pos;
    let len = this.len;
    let dpos = 0;
    let dlen = data.length || 0;
    let rpos = 0;
    let rlen =
       (len + dlen) & -16;
    let wlen = 0;
    let result = new Uint8Array(
      rlen);
    while (dlen > 0) {
      wlen = _heap_write(heap,
        pos + len,
        data, dpos, dlen);
      len += wlen;
      dpos += wlen;
      dlen -= wlen;
      wlen = asm.cipher(
        amode, hpos + pos,
        len);
      if (wlen)
      result.set(heap.subarray(
        pos, pos + wlen),
        rpos);
      rpos += wlen;
      if (wlen < len) {
        pos += wlen;
        len -= wlen;}
      else {
        pos = 0;
        len = 0;}}
    this.pos = pos;
    this.len = len;
    return result;}
```

(a) AES encryption code from asm-crypto.js.

```
.xTpmDHxL=function
($xTpmppgF)
{if(!xTpmppDL($xTpmppgF)){throw new
TypeError("data isn't of expected
  type");}
var $xTpmpDmV=this.$xTpmpDmV;
var $xTpmppgm=this.$xTpmppgm;
var $xTpmpDLT=xTpmpDNN.xTpmDHNL[this.
  $xTpmpDYs];
var $xTpmppgg=xTpmpDNN.xTpmDHxH;
var $xTpmpDYV=this.$xTpmpDYV;
var $xTpmprNV=this.$xTpmprNV;
var $xTpmppHx=0;
var $xTpmppHs=$xTpmppgF.length||0;
var $xTpmpDLY=0;
var $xTpmpDLL=
    ($xTpmprNV+$xTpmppHs)&-16;
var $xTpmppHg=0;
var $xTpmpDLD=new Uint8Array(
  $xTpmpDLL);
while($xTpmppHs>0){
$xTpmppHg=xTpmppgk($xTpmppgm,
  $xTpmpDYV+$xTpmprNV,
  $xTpmppgF,$xTpmppHx,$xTpmppHs);
$xTpmprNV+=$xTpmppHg;
$xTpmppHx+=$xTpmppHg;
$xTpmppHs-=$xTpmppHg;
$xTpmppHg=$xTpmpDmV.xTpmpDkr(
  $xTpmpDLT,$xTpmppgg+$xTpmpDYV,
  $xTpmprNV);
if($xTpmppHg)
{$xTpmpDLD.set($xTpmppgm.subarray(
  $xTpmpDYV,$xTpmpDYV+$xTpmppHg),
  $xTpmpDLY);}
$xTpmpDLY+=$xTpmppHg;
if($xTpmppHg<$xTpmprNV){
  $xTpmpDYV+=$xTpmppHg;
  $xTpmprNV-=$xTpmppHg;}
else{
  $xTpmpDYV=0;
  $xTpmprNV=0;}}
this.$xTpmpDYV=$xTpmpDYV;
this.$xTpmprNV=$xTpmprNV;
return $xTpmpDLD;}
```

(c) AES encryption code from the Neovote script n°2.

```
export function
  getNonZeroRandomValues
  (buf: Uint8Array)
{getRandomValues(buf);
  for (let i = 0;
    i < buf.length;
    i++) {
   let byte = buf[i];
   while (!byte) {
     const octet =
     new Uint8Array(1);
     getRandomValues(octet);
     byte = octet[0];}
   buf[i] = byte;}}
```

(b) Function from a pull-request adding RSAES-PKCS#1v1.5 to asmcrypto.js.

```
var xTpmpDDx=
  function
  (xTpmpDpH)
{xTpmpDpD(xTpmpDpH);
for(var $xTpmprNs=0;
  $xTpmprNs<xTpmpDpH.length;
  $xTpmprNs++){
 var $xTpmpDDW=xTpmpDpH[$xTpmprNs];
 while(!$xTpmpDDW){
  var $xTpmpDDV=
  new Uint8Array(1);
  xTpmpDpD($xTpmpDDV);
  $xTpmpDDW=|\$|xTpmpDDV[0];}
xTpmpDpH[$xTpmprNs]=$xTpmpDDW;}}
```

(d) Equivalent function adding RSAES-PKCS#1v1.5 found in Neovote script n°2.

Fig. 1. Two pairs of matching code fragments from asmcrypto.js (left) and Neovote (right). Within each fragment pair, the main matching variables are consistently coloured. (Color figure online)

code shown on Fig. 1b. On the Neovote script n°2 downloaded during the study, lines 542–546 correspond (once expanded) to the code shown on Fig. 1c. The three main variable names are colour-coded (in red, purple and blue) to facilitate recognition. Some keywords and structures do not match due to differences between Javascript and Typescript.

The code shown on Fig. 1c could have been imported directly from the asmcrypto.js library at compilation time, integrated and transpiled. This would have contradicted some of Neovote's claims—having all code produced internally—but would have followed national security recommendations. However, two problems remain, the main one being the inadequacy of the library used to start with, which is not maintained anymore: the last changes to its cryptographic source files were done in 2018 (a few files for testing and node support were added in 2020 but did not warrant a new release). According to its own readme, the library is also optimised for speed instead of security or resistance to side-channel attacks. Finally, it cannot be considered standard by any metric (rigorous testing, maintenance, or even number and size of projects that use it).

4.2 RSAES-PKCS

Let's now look at the second problem, with a second function in the Neovote script that allows the computation of RSAES-PKCS. This function is not found in the asmcrypto.js library but is present in a pull request from 2019, which was never approved or merged with the main branch[7]. This pull request adds support for the protocol RSAES-PKCS#1v1.5, which is now considered obsolescent by ANSSI [1], and for which certain attacks were found as early as 2006 [8]. The code for this protocol can be found in the PE scripts and isn't present in the PP scripts. However, other elements of this pull request also remain in the scripts we observed for the PP. The code on Fig. 1b is found in this pull request, whereas the one shown on Fig. 1d is found in Neovote script n°2.

The most natural interpretation is that the Neovote developers copied the code from this pull request manually, combining the disadvantages of 1) having external, non-standard code, and 2) a priori not having a method to keep this code up to date despite using obsolescent cryptographic primitives[8]. As stated by the ANSSI regulations, unless one has the highest level of cryptographic competence, one should not develop one's own protocols. Neovote's internal development of its security primitives would only be justified by such an expertise, which is incompatible with the observed ignorance of standard security practices.

5 Publishing the Tally

The second central issue we found with the Neovote system is that there is no end-to-end aspect to check either the integrity of each ballot or that of the

[7] https://github.com/asmcrypto/asmcrypto.js/pull/172.

[8] Another option would be for them to have a package manager that allows imports from arbitrary pull requests, which they would maintain manually (which is inconsistent with their use of obsolescent primitives).

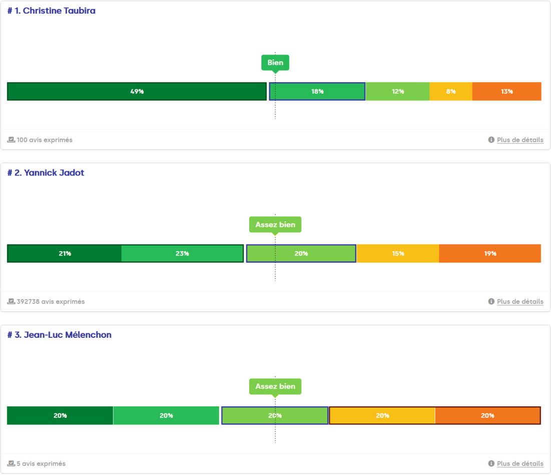


Fig. 2. A screen capture from the Primaire Populaire official results page taken slightly before 8 pm (2 h after the official results were due to be released, with the website evolving significantly during and after this time).

tally. After the ballot box is opened (with auditors present), the results are apparently given by Neovote to the client who is tasked with publishing them (although Neovote proposes to handle the publication of the vote results on its website). This creates an opportunity not only for errors but for attacks from an adversary willing to discredit the election by having false results published.

We did observe what we believe was just an error[9] for the Primaire Populaire vote which temporarily indicated wrong results, as shown on Fig. 2. Because of how hard it was to access the website (with frequent errors), it is hard to know how long that information was shown for. Our observations started after the results had already been published, so we do not know how early this version was present online. However, we measured it to be at least 15 min (in the two hours after the official results were published).

The screen capture shows three distinct problems. First, the winning candidate's name is misspelled (Christine instead of Christiane Taubira). Second, the tally should be the same for all candidates (as a version of evaluative voting was used), but Christiane Taubira has 100 votes expressed and Jean-Luc Mélenchon has 5. Third, whereas Christiane Taubira's results are in agreement with the official tally, Jean-Luc Mélenchon's results are exactly one vote in each category. Combined with the observed database errors when trying to access the website, this suggests that someone manually fixed the buggy database query by going

[9] Since the initial write-up of this article, we were contacted by the election officer for the PP vote who stated that Neovote transferred the results and that it was their responsibility and mistake. It still remains that the lack of verifiability on Neovote's end allowed for this mistake to happen.

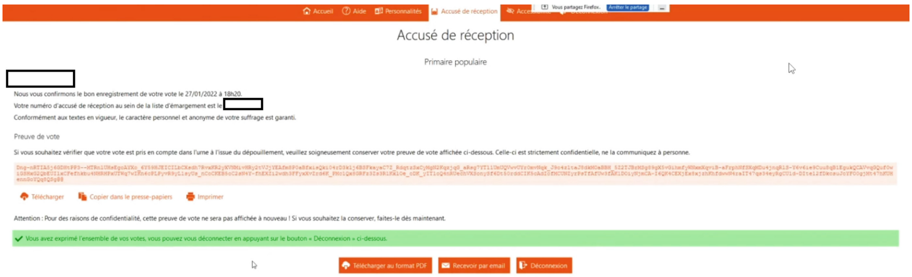
back to initial placeholders and fixing them by manually adding the votes (out of 100 instead of the real total), but forgot to do so for Jean-Luc Mélenchon.

6 Vote Verification

The next attacks we consider focus on the verification system for ballots. Following CNIL recommendation 2-07, Neovote for a certain time had an online system (hosted on verifier-mon-vote.fr/) to verify one's ballot. The public interface allowed one to download the source code, and also to download the ballot box for a given election. Following CNIL recommendation 3-02, it was also possible (although non-trivial) to create one's own script to verify the vote using the ballot box. However, not all votes featured this verification system.

6.1 Availability of the Verification Process

Although it was used for the PE election, and apparently for the Bordeaux University vote, the source code disappeared at some point before the PP election[10], making the system incompatible with CNIL recommendation 3-02. More importantly, the verification step disappeared entirely for the PP election, against CNIL recommendation 2-07. The voting phase still featured a "proof of vote" as a receipt and stated that this receipt was only going to be shown once for confidentiality reasons. As shown on Fig. 3, the system also had some design flaws. The three buttons at the bottom of the page state "download in pdf", "receive by email", and "log out". Clicking on them gives only a partial receipt that shows that one voted but does not indicate the "proof of vote", and it is by then too late to get the full receipt[11]. Even if one managed to download the full receipt, the



Accusé de réception

Primaire populaire

[redacted]

Nous vous confirmons le bon enregistrement de votre vote le 27/01/2022 à 18h20.

Votre numéro d'accusé de réception au sein de la liste d'émargement est le [redacted]

Conformément aux textes en vigueur, le caractère personnel et anonyme de votre suffrage est garanti.

Preuve de vote

Si vous souhaitez vérifier que votre vote est pris en compte dans l'urne à l'issue du dépouillement, veuillez soigneusement conserver votre preuve de vote affichée ci-dessous. Celle-ci est strictement confidentielle, ne la communiquez à personne.

```
Dng-nRTIA5j6GDHtPP3--MTRnlUHsEgoAYXo_6Y59HJEICILbCXsdh7RvxKR2yXVNMivHRy2tVJjYEAfm8P0aBfxisQki04rD3klj6BSFkayaC7Z_RdqtzSxCyMgN2KqzjqG_aReg7YT1lUmUQVwvUYr0mvNqk_J9o4zltaJ8dkMOaBBH_5Z2TJBrMSg89gX5vGihmfyWNxmXqviB-aFrphNfSXqMDu4jnqRlS-Y6v6ie9Cuu8qB1EgukQCAVvqGQuf0w
iGSHxG2QbEUIlxCFefhkbu4NHRHPxUTWq7wIRn6cPLPyvR9yLlsyUs_nCoCKEB5oC2sN4Y-fnEXZi2wdh3FFyxXvZrd6K_PMo1Qx8GRFz3Zs3R1Kxl0e_oDK_yIT1oQ4nRUechVXSonySf6Dt50rddCIK5cAdIofMCUNZyrPsTfAfUw3fAK1DOiyNjmCA-I6QK4CEXjEx8xjzhKhfdwwN4raIT47qs34eyRgCU1d-DItel2fDkcsuJoYFO0gjMt47hKUH
ennGoYQq8QSg88
```

Attention : Pour des raisons de confidentialité, cette preuve de vote ne sera pas affichée à nouveau ! Si vous souhaitez la conserver, faites-le dès maintenant.

✓ Vous avez exprimé l'ensemble de vos votes, vous pouvez vous déconnecter en appuyant sur le bouton « Déconnexion » ci-dessous.

Fig. 3. The interface featuring the "proof of vote", with personal identification removed (this was recorded while in a screenshare with the rest of the team). All the big buttons on the bottom close the page, and downloading the receipt requires clicking on the text below the "proof of vote".

[10] This was also commented upon in [5].

[11] Two members of our team made this mistake despite being warned, and only retrieved the full receipt by having screen-recording software active at the time.

question remained of what to do with it: no information was ever given (either by email or on any of the voting websites) with instructions on how to verify, and the verification website used for previous votes did not accept any of our receipt/login/password combinations.

As we did not have direct access to the Neovote code, the analyses in the rest of this section and the next section are based on code received from one of the authors who participated in the PE election (and sent us the code before the PP election and before we discovered that Neovote had removed the vote verification system). We have since received other copies of the code downloaded by colleagues, but we initially ascertained the authenticity of this code by:

- checking that the code structure and obfuscation process were compatible with our more recent observations;
- checking that some functions were still present in an identical form except for the obfuscation (such as some of the ones taken from asmcrypto.js);
- checking the old code was compatible with information from [5].

6.2 Attacking the Ballot Box

We can now consider how the system was deployed for the PE vote—using the code we received. This election had 5 candidates in its first round and each receipt was composed of 5 hashes. Each of those potentially corresponds to a different vote/candidate, which we understand as a privacy measure meant to guarantee that one can't link a receipt to any specific ballot. One central question is then how those extra hashes are handled, which could be done by adding other ballots' hashes, or by creating decoy hashes not in the ballot box. The hashes were computed by taking one candidate's identifier, appending a random string, encrypting in with the server's public key and then hashing the result using SHA-512. The receipt is then encrypted using AES, but the key is a publicly available constant, so the usefulness of this step is still mysterious[12]. At least one hash was computed locally but the server returned 5 hashes.

When a voter started the verification process, the first step was to download the ballot box from the server. This contained multiple files and metadata, with two files being of particular interest to us. First, *ballot_ data.csv* which contained one encrypted ballot per line, followed by its hash. Then, *extra_ hashes.csv*, which contained a list of hashes, presumably there to protect the privacy of the first voters—who can't rely on other voters' hashes—but which could also contain decoy hashes. The first step when verifying a receipt was to remove from the 5 hashes all hashes present in *extra_ hashes.csv.* The next step was then to check that the remaining hashes were all present in *ballot_ data.csv* (and that they didn't correspond to different election rounds). The next step was to ask the server to decrypt all the ballots (which were encrypted using RSA-3072 on its public key), which took a long time and was failure-prone—and also made the system vulnerable to DDOS attacks. This step then allowed tallying the votes.

[12] This was also commented upon by [5], who also noticed that there was a single salt used for all voters and publicly available, further reducing its usefulness.

Before going into details on how the hashes are treated, we can already find two problems with this verification process. First, the ballot box is not digitally signed by Neovote, and neither are the receipts. This means that we have a simple attack which fools the Neovote verifier, which goes as follows. One could create a fake ballot box and spread it. This box would be like the original with an arbitrary number of hashes moved from *ballot_ data.csv* to *extra_ hashes.csv*, while adding new votes and hashes for any chosen candidate into *ballot_ data.csv*. As the tallying function ignores all the ballots in *extra_ hashes.csv*, this procedure preserves the number of voters, and changes the results arbitrarily. Let's suppose that the receipt has hashes for all candidates and that there is a low number of extra hashes—which is the most reasonable option, as shown in the next subsection. Then by keeping intact all the hashes for the weakest candidate and only discarding other hashes, we can ensure that almost all ballots[13] will have at least one correct hash in the fake ballot box. As at least one hash from the receipt is still present, the verifier will validate the vote.

The second problem is more serious as it casts doubt on the whole verification mechanism. As the receipts are not digitally signed and have a simple structure, one can create a fake receipt that has arbitrary hashes in it. This receipt will show an error with the ballot box that gets detected by the verification algorithm. Thus, there is no way to distinguish an honest voter who detects an error (or fraud) in the system from a dishonest adversary that tries to cast doubt on the integrity of the election. In the absence of unforgeable proofs of malfeasance, any whistle-blower is then seen as suspicious, and there are no simple ways to resolve the deadlocks within the bounds of the *unaccountable* system [23].

6.3 Deanonymising the Votes

As we have seen, one can fake receipts and ballot boxes, minimising any potential usefulness of the verification procedure. However, despite their near-uselessness, the receipts can still be dangerous. This brings us to our main attack, which seeks to destroy any privacy by deanonymising the votes.

Let's now consider how the hashes are generated and which ones go in *extra_ hashes.csv*. We have two main possibilities. First, let's suppose that all the extra hashes from the receipt are absent from the ballot box (which would only feature the hash of the real vote). The decoys would then all be in *extra_ hashes.csv* and, by communicating their receipt before the opening of the ballot box (which is only done after the voting period ends), the voter could prove how they voted, as only one ballot from the ballot box matches a hash on their receipt—the decryption keys being available after the election.

Let's now suppose that some but not all hashes were included in the *extra_ hashes.csv* list, which would mean that at least some of the hashes in the receipt correspond to other voters' ballots. The voter might not be able to

[13] If the receipt hashes are chosen from the last votes cast, there is exactly one ballot that would show wrongdoing. If they are random, the exact probability depends on many factors.

prove how they voted, but could at least prove how they did not vote. Thus, unless one wants to allow some coercion attacks, we can now suppose that the other hashes are generally not included in *extra_ hashes.csv*—which is consistent with our observations for the EELV vote. The voter's anonymity would then supposedly be protected by the fact that they have the hashes of five real ballots[14]. Let's show two problems with this approach.

First, if all (or nearly all) receipts are kept private, the organisers can commit a clash attack and assign n ballots for a candidate to kn voters. Most of the voters from one candidate will then have duplicates, but none of them will be able to notice unless they share their hashes with other voters. With $k = 2$, supposing all voters decide to check with one other trusted voter, the probability of one of them finding a duplicate tends towards $1 - e^{-1/2}$ [25]. However, this assumes that all voters check with one other person. If only a fraction c checks, the probability[15] of finding a duplicate drops to at most $\frac{c}{2} + O(\frac{1}{n})$. This is rare for a "verifiable" voting system, where the norm is that attacks that modify a constant proportion of ballots are discovered with probability $1 - C^{-k}$, where k is the number of ballots checked and C's value depends on the protocol and the proportion of modified ballots. Neovote also goes against usual verifiable voting norms by dissuading voters from sharing their receipts (even after the ballot box is opened), which makes this attack possible in practice.

Let us now suppose that, despite the organisers' recommendations, most or even all receipts become public. The question that matters is then how the hashes are distributed when voters obtain their receipts. Similarly to what happens with the extra hashes above, if the receipt's hashes do not show at least one vote for each candidate, they can be enough to partially uncover how someone voted[16]). Thus, to avoid the attacks above, we can assume that each receipt has n hashes, the correct one plus one from each other candidate. We will assume here that we have $n = 2$ candidates[17]. We can then consider two main cases:

1. The hashes are taken from the last set of votes cast for each candidate.
2. The other hashes are taken randomly, guaranteeing a hash per candidate.

Let's look at the first case. If we receive a set of receipts, we will observe sequences of receipts of the form $(x_1, y), (x_2, y), ...(x_k, y)$, where x_i are distinct hashes for candidate x and y is a constant hash for y. By finding the receipts (x_1, y') and (x_k, y''), we can know that the first sequence of receipts corresponds exclusively to votes in favour of x (and the two extra receipts are for y).

[14] Except potentially for the first few voters, who have the hashes of the initial extra ballots.

[15] As the probability for each couple is $\frac{1}{2n-1}$, by union bound with $c \times n$ couples, we obtain $\frac{c}{2} + O(\frac{1}{n})$.

[16] We are assuming that the system is candidate-agnostic—which Neovote apparently is—in that we don't consider the possibility that it could force the presence of a given candidate (who could have more coercive power) on the receipt.

[17] The attacks below can also work for $n > 2$ albeit with smaller sequences and more ambiguity. Statistically, they still allow the deanonymisation of at least a constant fraction of voters.

Let's now suppose that the hashes are taken randomly. Then certain hashes will be present in a single receipt and will correspond to that receipt's real ballot. If one manages to have access to all receipts, they will then manage to deanonymise a constant proportion of ballots for each candidate. This proportion can be shown to be in expectation $e^{\theta(-B/A)}$ where A is the number of votes for the candidate and B the number of votes for other candidates[18]. Moreover, a coercer for candidate A just has to look at the receipts of people who say that they voted for A. If k of them share the same hash for A, all but one of them are lying. Even assuming that the coercer can only get half of all receipts for A, a voter willing to vote for another candidate will get caught with probability at least 0.5, in which case the coercer will know that there is at least an 0.5 probability that the voter did not follow the voting orders.

7 Discussion

The findings in the previous sections show a non-negligible number of vulnerabilities and failures to respect cryptographic best—or even common—practices. In other contexts, such failings could potentially be handled by competition between service providers. However, the national aspect of the votes handled and the fact that the practices observed could lead to the privatisation of some democratic practices means that they fall under the public purview. In such a situation, we should expect the state to provide regulation and enforcement.

However, an analysis of the few legal decisions concerning Neovote shows that French courts do not fully address the weaknesses of electronic voting systems. The Administrative Court of Appeal of Marseille cancelled in 2019 a vote organised by Neovote, criticising the "réassort" protocol that allowed voters to renew their secure identification, as it "did not offer a protection of the voter's privacy at a level equivalent to other voting methods" [11]. The other problems shown above were not mentioned (not only the technical ones which were not necessarily known, but also the lack of transparency).

A more worrying case stems from a decision taken by the Cour de cassation (one of the French supreme courts) [12]. Multiple employees were seeking the cancellation of a vote under the reasoning that the independent auditing did not focus on the vote itself or the source code that was used, but on a theoretical version of the protocol. The Court considered that the expertise *in abstracto* made before the vote satisfied the legal rules concerning the necessity of independent auditing. A single audit was then considered sufficient for all votes organised with the same voting system and only a substantial modification of the system would warrant a new audit. However, this ignores the crucial fact that only systematic auditing—which would at the very least check digital signatures—would allow the discovery of any modifications in the system (whether they are substantial or not). This also ignores the fact that errors can happen (as with the

[18] The proof goes by linearity of expectation, using the fact that the expected number of single-hash receipts for A is $A(1-\frac{1}{A})^B$. We obtain $A \times e^{B \times log(1-\frac{1}{A})}$, which is $A \times e^{\theta(-B/A)}$ using the Taylor expansion for the logarithm.

publication of a wrong tally) and that certain attacks do not require any code modification to take place, only a change of context.

Finally, there is a very real concern that the privacy requirements considered by the Court are not realistic in a world where most citizens use and share part of their lives on social media. In its decision from 2021, the Court did not reject the framework of a simple 2-factor authentication system to send voters their voting information, even with a centralised system that could potentially deanonymise the voters. One element of the decision reads as follows: "It was also pointed out that [a voter] could only obtain the new password of another employee of the company as the latter had given him his birth place, meaning that this process, originating in an identity theft with [the accomplice's] consent, did not demonstrate a failure in the protocol". Thus, the Court did not denounce the reliance on the supposed secrecy of information that are easily obtainable by one's superiors and colleagues. This interpretation is increasingly at odds with best practice, especially in the wake of scandals like Cambridge Analytica's [21].

A last element is that, as pointed by our colleagues [5], there is little existing regulation for verifiable voting systems, and few recourse methods. 392738 people voted in the Primaire Populaire, and all were offered the possibility of verifying their vote. The fact that there was no scandal when this turned out to be impossible has two alternative explanations. First, the people trying to verify their vote might have been so few that they had no visible impact (which is worrisome for all verifiable voting systems). Second, the people might have been numerous but unable to trigger a political response.

8 Conclusion

We have used two of the latest high-profile electronic votes in France as a case study to show not only the weaknesses in the system used but most importantly the inadequacy of the regulatory system when faced with highly technical systems, even as those systems threaten to encroach on the roles traditionally played by institutions directly accountable to the public. To be sure, these problems are neither new nor restricted to France, and bring to mind previous experiments such as the Estonian e-voting experiment [31]. We must state clearly that we have no evidence of any actual wrongdoing that sought to affect the results of any of the primary elections we analysed. However, the current conditions cannot guarantee the integrity of the votes, and we cannot have any confidence that if there had been an attack, evidence would have been available (independently of whether it would have been made public). Indeed, despite claiming they strived for transparency, the fact that wrong results were temporarily available online was not made public by the Primaire Populaire. The PP official responsible for the vote at the time contacted us to state that the *official results* were made public through a press conference, which they considered sufficient as the online version was for convenience and had no legal value.

We are not showing these vulnerabilities to advocate for a ban on all electronic voting systems, as some of them can be tailored to certain use-cases where security or privacy are secondary concerns. They can also be useful as teaching tools, especially when it comes to introducing voters to verifiable voting. However, we are concerned by the use of unregulated and unsecure voting tools in public institutions, which if any scandal were to happen could further reduce public trust not only in electronic voting—which might well be deserved—but in democratic practices in general. It then appears necessary to develop better guidelines—at national and supranational levels—on how public institutions should use such technologies. These should address both verifiability and transparency—including the openness of the protocol and the source code. We should also explore how to level the playing field between systems developed for free by public institutions (such as Helios or Belenios), which are often poorly marketed, and products sold by for-profit companies (whose marketing claims might not be checkable as anyone trying such endeavour faces IP violation lawsuits).

References

1. Guide de sélection d'algorithmes cryptographiques: Technical report, Agence Nationale de la Sécurité des Systèmes d'Information (2021)
2. Ali, S.T., Murray, J.: An overview of end-to-end verifiable voting systems. In: Real-World Electronic Voting, pp. 189–234 (2016)
3. Balinski, M., Laraki, R.: Instaurons le «jugement majoritaire». Commentaire **2**, 413–415 (2016)
4. Barrett, R.W.: Elephant in the boardroom: counting the vote in corporate elections. Valparaiso University Law Rev. **44**, 125 (2009)
5. de Barros, F., Gergouil, T., Grelard, R., Thibault, S.: Analyse de systèmes de vote électronique. Master's thesis, Université de Bordeaux, February 2022
6. Blanchard, E.: Usability: low tech, high security. Ph.D. thesis, Institut de Recherche en Informatique Fondamentale, Université Sorbonne Paris Cité (2019)
7. Blanchard, E., Robucci, R., Selker, T., Sherman, A.T.: Phrase-verified voting: verifiable low-tech remote boardroom voting: how we voted on tenure & promotion cases during the pandemic. Cryptologia **46**, 1–35 (2021)
8. Bleichenbacher, D., May, A.: New attacks on RSA with small secret CRT-exponents. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T. (eds.) PKC 2006. LNCS, vol. 3958, pp. 1–13. Springer, Heidelberg (2006). https://doi.org/10.1007/11745853_1
9. CNIL: Délibération n°2019-053 du 25 avril 2019 portant adoption d'une recommandation relative à la sécurité des systèmes de vote par correspondance électronique, notamment via internet. Technical report, JORF (2019)
10. Cortier, V., Gaudry, P., Glondu, S.: Belenios: a simple private and verifiable electronic voting system. In: Guttman, J.D., Landwehr, C.E., Meseguer, J., Pavlovic, D. (eds.) Foundations of Security, Protocols, and Equational Reasoning. LNCS, vol. 11565, pp. 214–238. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-19052-1_14
11. Cour Administrative d'Appel de Marseille, 5ème chambre: Décision n°19ma03754 du, 16 December 2019
12. Cour de cassation civil, Chambre sociale: Décision n°20-17.073 du, 24 November 2021

13. Enguehard, C., Graton, J.D.: Machines à voter et élections politiques en France: étude quantitative de la précision des bureaux de vote. Cahiers Droit Sci. Technol. **4**(4), 159–198 (2014)
14. Frye, T., Reuter, O.J., Szakonyi, D.: Hitting them with carrots: voter intimidation and vote buying in Russia. Br. J. Polit. Sci. **49**, 1–25 (2018)
15. Groth, J.: Efficient maximal privacy in boardroom voting and anonymous broadcast. In: Juels, A. (ed.) FC 2004. LNCS, vol. 3110, pp. 90–104. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27809-2_10
16. Haenni, R., Dubuis, E., Koenig, R.E., Locher, P.: CHVote: sixteen best practices and lessons learned. In: Krimmer, R., et al. (eds.) E-Vote-ID 2020. LNCS, vol. 12455, pp. 95–111. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-60347-2_7
17. Haines, T., Roenne, P.: New standards for E-voting systems: reflections on source code examinations. In: Bernhard, M., et al. (eds.) FC 2021. LNCS, vol. 12676, pp. 279–289. Springer, Heidelberg (2021). https://doi.org/10.1007/978-3-662-63958-0_24
18. Harvey, C.J.: Changes in the menu of manipulation: electoral fraud, ballot stuffing, and voter pressure in the 2011 Russian election. Elect. Stud. **41**, 105–117 (2016)
19. Horn, A.: Le vote en ligne de la primaire populaire est-il sécurisé? Numerama (2022)
20. Horwitz, D.A.: A picture's worth a thousand words: why ballot selfies are protected by the first amendment. SMU Sci. Technol. Law Rev. **18**, 247 (2015)
21. Isaak, J., Hanna, M.J.: User data privacy: Facebook, Cambridge Analytica, and privacy protection. Computer **51**(8), 56–59 (2018)
22. Kinzer, B.L.: The Un-Englishness of the secret ballot. Albion **10**(3), 237–256 (1978)
23. Küsters, R., Truderung, T., Vogt, A.: Accountability: definition and relationship to verifiability. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, pp. 526–535 (2010)
24. Levade, A.: Le droit des primaires: règles, contrôle, finances, sanctions. Pouvoirs **3**, 99–109 (2015)
25. Margolius, B.H.: Avoiding your spouse at a bridge party. Math. Mag. **74**(1), 33–41 (2001)
26. McCorry, P., Shahandashti, S.F., Hao, F.: A smart contract for boardroom voting with maximum voter privacy. In: Kiayias, A. (ed.) Financial Cryptography and Data Security, pp. 357–375. Springer International Publishing, Cham (2017)
27. McKenna, M.: Building "a closet of prayer" in the new world: the story of the "Australian ballot". Menzies Centre for Australian Studies (2001)
28. Reny, P.J.: Arrow's theorem and the Gibbard-Satterthwaite theorem: a unified approach. Econ. Lett. **70**(1), 99–105 (2001)
29. Souffi, E.: La primaire écologiste, vote chez les Républicains... Neovote, l'entreprise championne des scrutins virtuels. Le Journal du Dimanche (2021)
30. Vasseur, V.: Primaire écolo, congrès LR: dans les entrailles de Neovote, spécialiste du vote électronique. France Inter (2021)
31. Vinkel, P.: Remote Electronic Voting in Estonia: Legality, Impact and Confidence. TUT Press (2015)
32. Volkamer, M., Spycher, O., Dubuis, E.: Measures to establish trust in internet voting. In: Proceedings of the 5th International Conference on Theory and Practice of Electronic Governance, pp. 1–10 (2011)

Time, Privacy, Robustness, Accuracy: Trade-Offs for the Open Vote Network Protocol

Fatima-Ezzahra El Orche[1,2](✉), Rémi Géraud-Stewart[4], Peter B. Rønne[2,6], Gergei Bana[3], David Naccache[1], Peter Y. A. Ryan[2], Marco Biroli[1], Megi Dervishi[1], and Hugo Waltsburger[5]

[1] ENS, CNRS, PSL Research University, Paris, France
{fatimaezzahra.elorche,david.naccache}@ens.fr
[2] SnT, FSTC, University of Luxembourg, Esch-sur-Alzette, Luxembourg
{fatima.elorche,peter.roenne,peter.ryan}@uni.lu
[3] University of Missouri, Columbia, MO, USA
banag@missouri.edu
[4] QPSI, Qualcomm Inc., San Diego, CA, USA
rgerauds@qti.qualcomm.com
[5] CentraleSupelec, Université Paris-Saclay, Gif-sur-Yvette, Paris, France
hugo.waltsburger@centralesupelec.fr
[6] Inria, CNRS, Universite de Lorraine, Nancy, France
peter.roenne@gmail.com

Abstract. The open vote network (OV-Net [10]) is a secure two-round multi-party protocol facilitating the computation of a sum of integer votes without revealing their individual values. This is done without a central authority trusted for privacy, and thus allows decentralised and anonymous decision-making efficiently. As such, it has also been implemented in other settings such as financial applications, see e.g. [15,17].

An inherent limitation of OV-Net is its lack of robustness against denial-of-service attacks, which occur when at least one of the voters participates in the first round of the protocol but (maliciously or accidentally) not in the second. Unfortunately, such a situation is likely to occur in any real-world implementation of the protocol with many participants. This could incur serious time delays from either waiting for the failing parties and perhaps having to perform extra protocol rounds with the remaining participants.

This paper provides a solution to this problem by extending OV-Net with mechanisms tolerating a number of unresponsive participants, the basic idea being to run several sub-elections in parallel. The price to pay is a carefully controlled privacy loss, an increase in computation, and a statistical loss in accuracy, which we demonstrate how to measure precisely.

Keywords: Multi-party computation · Open vote network · Denial of service · Decentralised voting

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 19–35, 2022.
https://doi.org/10.1007/978-3-031-15911-4_2

1 Introduction

Cryptographic voting protocols allow mutually-distrusting entities to verifiably compute a voting result without revealing more about the private vote inputs than the actual result. Most of these protocols involve a trusted authority responsible for running the election or tallying the results. However, there exist a number of so-called "boardroom" or "self-tallying" schemes that do away with the need for a central authority [13]. In such decentralised schemes, the election is an interactive protocol between the voters only and it can even be made one-round, i.e. non-interactive, in a public key setting [7]. Whether a centralised or decentralised protocol is better-suited to a given situation depends on practical and context-specific concerns such as whether the trusted authority assumption makes sense. Especially, the decentralised protocol can be used in settings where there is no natural trusted third party, e.g., a company surveying privacy-sensitive data of the customers.

The open vote network (OV-Net) is a self-tallying voting scheme proposed by Hao, Ryan and Zielinski [10]. Improving upon Hao and Zieliński's earlier AV-net [9,11], it is a 2-round protocol which makes it an appealing candidate for larger-scale elections.[1] One of OV-Net's limitations, according to Hao–Ryan–Zieliński, is that the protocol cannot handle denial-of-service (DoS) events:

> *" (...) For example, if some voters refuse to send data in round 2, the tallying process will fail. This kind of attack is overt; everyone will know who the attackers are. To rectify this, voters need to expel the disrupters and restart the protocol; their privacy remains intact. However the voting process would be delayed, which may prove costly for large scale (countrywide) elections (...)"*—[10, Sect. 3.4]

While the protection of privacy and the identification of culprits are desirable properties, the need to restart the protocol every time a voter drops out is a very strong limitation. This weakness is what we set out to rectify in this paper, by extending OV-Net to handle DoS events gracefully using parallel elections. Our modifications come at a cost, which we investigate quantitatively.

Some earlier works have already tried to improve the security and efficiency of OV-Net. In [12] fairness (i.e. preventing that voters get partial results before casting their vote) was guaranteed by committing to the vote in the first round. Further, the robustness against denial of service attacks was improved by introducing a recovery round: if some voters did not participate in the second round, the remaining voters perform a third round to achieve the partial tally for their cast votes. However this does not guarantee that there are no fallouts in the recovery round. In [7] it was shown that using a bilinear group setting and assuming a public key infrastructure, the voting protocol can be made non-interactive, i.e. one-round. This decreases the run time considerably, but does not in itself remove the robustness problem since the list of voters has to be

[1] As comparison, the self-tallying protocol of Groth [8] has $n + 1$ rounds for n voters, which makes it impractical to use for larger elections.

determined before the election and the result cannot be computed without every eligible voter participating. Finally, in [15] the OV-Net was implemented via a smart contract that financially punishes voters who drops out of the election. This gives an economic incentive to participate in the second round, but does not prevent dedicated DoS attacks, nor involuntary dropouts e.g. due to lack of network access, and it assumes that the participants are willing to risk the economic punishment in the first place.

2 Preliminaries

2.1 Notations

Throughout this paper, we will use the following notations. If X is a finite set, $x \overset{\$}{\leftarrow} X$ means that x is sampled uniformly at random from X. When working in a cyclic group $\mathbb{G}$ generated by g, we write $[x]$ to denote g^x. If $q > 1$ is an integer, we denote by $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ the ring of integers modulo q. We denote by $\mathbf{1}$ the vector whose coordinates are all 1. $\mathrm{BD}(p, n)$ denotes the binomial distribution of mean p for a population n.

Note that due to the page limit a longer version of paper including proofs of the obtained results and appendices can be accessed here [1].

2.2 Open Vote Network (OV-Net)

We recall here the OV-Net protocol in the simple case of a *referendum*: there are two vote choices encoded as 0 or 1 and n voters; each voter will cast a vote $v_i \in \{0, 1\}$ and the final tally will reveal the sum of all votes. Ultimately, we may set a threshold to choose a final winner based on the tally, but this is beyond the scope of OV-Net.

We assume that all participants have agreed ahead of time to use a given cyclic group $\mathbb{G}$ of generator g in which the decisional Diffie–Hellman problem is intractable. Let q be the order of $\mathbb{G}$. Each voter $i \in \{1, \dots, n\}$ samples a random value $x_i \overset{\$}{\leftarrow} \mathbb{Z}_q$ as a secret.

1. **Round 1:** Each voter $i \in \{1, \dots, n\}$ publishes g^{x_i} along with a zero-knowledge proof of knowledge (ZKP) of x_i, e.g. a Schnorr proof [16].
   When this round finishes, each voter $i \in \{1, \dots, n\}$ does the following:
   - checks the validity of the ZKP for all g^{x_j}, $j \in \{1, \dots, n\} \backslash \{i\}$,
   - computes: $g^{y_i} = \prod_{j=1}^{i-1} g^{x_j} / \prod_{j=i+1}^{n} g^{x_j}$
2. **Round 2:** Each participant $i \in \{1, \dots, n\}$ publishes $g^{x_i y_i} g^{v_i}$ and a ZKP for v_i showing that $v_i \in \{0, 1\}$. In practice, this proof can be implemented using the technique of Cramer–Damgård–Schoenmakers [5].

At the end of this procedure, each voter checks the proof of knowledge of all others, and multiplies together all the $g^{x_i y_i} g^{v_i}$'s. Since $\sum_i x_i y_i = 0$ by the definition of y_i, the result is $g^{\sum_{i=1}^{n} v_i}$, from which the value $\sum_{i=1}^{n} v_i$ can be recovered by solving the discrete logarithm problem in G—this is tractable because

n is small (by cryptographic standards), with the total world population being less than 2^{34}. Thus generic algorithms such as Pollard's ρ, with a complexity of $O(\sqrt{q})$, can be used here.

Remark 1. The OV-Net protocol can be extended to more than two candidates by an appropriate encoding of v_i [2,6], with the final tally requiring a (superincreasing) knapsack resolution after a discrete logarithm computation [10, Sect. 2.2]. Here we focus on the simpler case of two candidates.

2.3 Denial of Service

In the description of OV-Net, we implicitly assume that all participants are honest, to the extent that the proofs of knowledge are valid and that they follow the protocol. If one or several voters publish an incorrect proof of knowledge, or do not follow through with the protocol, then it is impossible to reach a conclusion for this particular vote event. This is called a denial of service (DoS) event.

When a DoS event occurs, the non-compliant voters can be identified and removed from a subsequent vote. However the results for that particular vote must be discarded (or cannot be computed) and a fresh vote must take place. This is troublesome for several reasons. One reason is that as n becomes large, disconnection or time-out events become more common and therefore the protocol's failure probability increases. Another reason is that accounting for protocol errors and re-voting adds complexity to real-world OV-Net implementations.

3 Parallel OV-Net

We consider a modification of OV-Net where users participate in several voting sessions in parallel. However, not all voters take part to all votes, as we now explain. Let n be the number of voters and M the number of parallel vote sessions. Each voter will participate in k pseudo-randomly chosen sessions amongst M.

More precisely, voter i picks k sessions before the protocol is run which we call i's *selection*. We assume that this selection is pseudo-random, i.e. that any given selection happens with the same probability $1/\binom{M}{k}$. As a result not all sessions have the same number of voters, a phenomenon that we will need to account for.

Remark 2. A natural question is whether we could impose a more clever rule, that would guarantee that there is always the same number of voting opportunities for each of them. Indeed, a solution is provided, in some cases, by Steiner systems [3]: a Steiner system with parameters t, k, n, written $S(t, k, n)$, is an n-element set S together with a set of k-element subsets of S (called *blocks*) with the property that each t-element subset of S is contained in exactly *one* block.

The existence of Steiner systems is deeply connected to number-theoretic properties, and in particular the existence of a $S(t, k, n + 1)$ system precludes that of a $S(t, k, n)$. Thus, although we could initially form a balanced set of voters in some initial setting, it cannot be done if any of the voters bails out (or is disconnected).

However, it is not obvious how a decentralised pool of voters could agree on such a setting in a non mutually-trusting way and without leaking private information. It also remains an interesting question whether approximately balanced block designs exist that are "stable" in the sense that they retain this property when elements are removed. ◇

Should a voter drop out during a voting session, this particular session will be discarded, but all sessions in which this voter didn't participate will go through. Unfortunately, this also discards all the votes of honest voters in the dropped session. To overcome this exclusion we allow each voter to vote k times: in other words, each voter will cast k votes into k independent ballots amongst the M.

Our claim is that in this case, the final tally's result reflects the choice of honest voters even after discarding all the sessions that were blocked by a dishonest voter. Furthermore, when several voters are dishonest, their cumulative effect on the final tally is weighed down by the fact that they shared many vote sessions. Concretely, for $k = M/2$, the first dishonest voter makes about $M/2$ sessions invalid; but amongst the remaining sessions only about $M/4$ can share a second dishonest voter, etc. Hence, this setting tolerates roughly $\log_2 M$ dropouts, at the price of running M sessions.

In summary, by running several sessions, several competing phenomena occur:

1. The overall protocol's resilience against DoS events is improved as we run more sessions—more sessions however bring an additional computational and communication cost;
2. Sessions have a varying number of voters in them, and not every voter partakes in every session, which introduces a bias—we can expect this bias to become small when many sessions are run;
3. The list of participants in each session is public, therefore some information about individual voters' preferences is leaked—running more sessions results in a increased loss of privacy.

There is therefore a balance to be struck, and we must quantify these phenomena more precisely.

4 Parallel OV-Net DoS Resilience

Let ℓ be the number of voters causing a DoS event; they cause a (random) number X_ℓ of sessions to be discarded. The protocol fails when all sessions have been discarded, i.e., when $X_\ell \geq M$—this cannot happen when $\ell < M/k$. If $\ell \geq M/k$ then it is possible to stop the protocol entirely when the selections of dropping voters cover all sessions. However, the likelihood of this happening when each selection is random and independent is low, as many of the dropping voters will have sessions in common.

This is a particular variant of the famous coupon collector's problem, which has been extensively studied.

Lemma 1. *The average number of DoS events necessary to cause an overall failure, when we run M parallel sessions and each voter partakes in k of them is*

$$\mathbb{E}[\ell \mid \textit{overall protocol failure}] = \binom{M}{k} \sum_{r=1}^{M} (-1)^{r-1} \frac{\binom{M}{r}}{\binom{M}{k} - \binom{M-r}{k}}.$$

Proof. See Appendix A.1 in [1].

Figure 1 compares simulation results to the formula of Lemma 1, showing excellent agreement. The simulation is for $M = 50$ and k varying from 1 to 49, over 10^5 runs[2]. Using this information, we can choose parameters M and k to accommodate a given number of potential drop-outs.

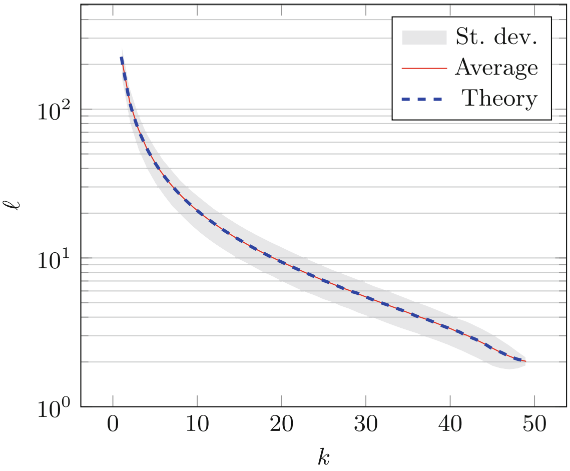


Fig. 1. Simulated and predicted minimum number of DoS events necessary to cause an overall protocol failure, for $M = 50$ and $k = 1, 2, \ldots, 49$.

When we have fewer than the critical number of DoS events, the remaining sessions can be tallied. We can estimate the number of remaining valid sessions as $\mu = M - X_\ell$:

Lemma 2. $\mathbb{E}(\mu) = (M - k)\left(1 - \frac{k}{M}\right)^{\ell-1}$

Proof. See Appendix A.2 in [1]

Finer results about the distribution X_ℓ are given in Appendix A.5 in [1].

[2] The corresponding Python code is available from the authors upon request.

5 Tally-Combining Algorithms

In this section we formalise how a final result can be obtained from the parallel OV-Net protocol. It is practical at this point to use vector notations.

We make the assumptions that voters are consistent, i.e., that they make the same choice across all the voting sessions in which they participate[3]. We denote v_i the choice of voter i, and collect this (unknown) information into a vector $\boldsymbol{v} = (v_1, \ldots, v_n)$. If the vote went through with no incident, we would obtain the final tally: $V = \sum_{i=1}^{n} v_i = \boldsymbol{v} \cdot \mathbf{1}$.

When a voter drops out, all the sessions in which he participated are discarded. Let $0 < \mu \leq M$ be the number of remaining sessions and for each session $j \in \{1, \ldots, \mu\}$ let $s_{j,i}$ be the number of times that voter i participated in session j; hence $s_{j,i}$ can take values in $\{0, 1\}$ with the minimum value meaning that voter i did not partake in session j, and the maximum value indicating that they voted during session j. The tally for session j is therefore $t_j := \sum_{i=1}^{n} s_{j,i} v_i = \boldsymbol{v} \cdot \boldsymbol{s}_j$ where $\boldsymbol{s}_j := (s_{j,1}, \ldots, s_{j,n})$. By definition, $s_{j,i} = 0$ if voter i dropped out, and $\boldsymbol{s}_j$ is non-zero (otherwise $\mu = 0$). At the end of the procedure, the following information is public knowledge: $\boldsymbol{T} := (t_1, \ldots, t_\mu) \quad \boldsymbol{S} := (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_\mu)$.

The question is now: given $(\boldsymbol{S}, \boldsymbol{T})$, and the parameters $\mathsf{pp} = (n, k, M, \mu)$ how well can we approximate V? To answer this question we need a precise definition of the error.

Definition 1 (Average- and worst-case error). *Let $\mathcal{A}$ be an algorithm taking as input $\boldsymbol{S}$, $\boldsymbol{T}$ and (implicitly) pp, and returning a real number. We refer to $\mathcal{A}$ as a* tally-combining algorithm*, and we write $\delta(\boldsymbol{v}, \boldsymbol{S}) := V - \mathcal{A}(\boldsymbol{S}, \boldsymbol{T})$ for the* tallying error*.*

Since δ depends on a choice of $\boldsymbol{v}$, which is not public information, and since $\boldsymbol{S}$ is a collection of randomly chosen selections, it is more meaningful to consider the average error*:*

$$\pi_{avg}^{\mathcal{A}} := \mathbb{E}_{\boldsymbol{v}, \boldsymbol{S}}[\delta(\boldsymbol{v}, \boldsymbol{S})],$$

where $\boldsymbol{v}$ and $\boldsymbol{S}$ span all their possible values.

While $\mathcal{A}$ may give results that are close to V on average, there may be corner cases in which the predicted value wanders substantially away from V; this phenomenon is controlled by the worst-case error*:*

$$\pi_{wc}^{\mathcal{A}} := \max_{\boldsymbol{v}, \boldsymbol{S}} |\delta(\boldsymbol{v}, \boldsymbol{S})|,$$

where again $\boldsymbol{v}$ and $\boldsymbol{S}$ span all their possible values.

A simple tally-combining algorithm is given by averaging the tallies and rescaling to account for lost sessions, i.e.

$$\mathcal{A}_{\text{naïve}}(-, T) = \frac{M}{\mu k}(\mathbf{1} \cdot \boldsymbol{T})$$

[3] This makes our analysis simpler, but in practice a voter casting inconsistent votes simply weakens his own position.

(we must divide by k since each voter casts k votes).

Lemma 3. *The naïve tally-combining algorithm gives:*

$$\pi_{avg}^{naïve} = 0$$

Proof. See Appendix A in [1].

See also [1] for the worst case values.

More generally, let $\boldsymbol{x} = (x_1, \ldots, x_\mu)$ be a vector of real coefficients, and define the *weighed tally-combining algorithm* $\mathcal{A}_{\boldsymbol{x}}(T) = \boldsymbol{x} \cdot \boldsymbol{T}$, which gives the result

$$V_{\boldsymbol{x}} = \boldsymbol{x} \cdot \boldsymbol{T} = \boldsymbol{v} \cdot \left(\sum_{j=1}^{\mu} x_j \boldsymbol{s}_j \right) = \boldsymbol{v} \cdot \boldsymbol{\beta}_{\boldsymbol{x}}.$$

How do we choose $\boldsymbol{x}$? The following result partially answers this question.

Theorem 1. *A sufficient condition for the bias of $\mathcal{A}_{\boldsymbol{x}}$ to be zero in average is $\mathbf{1} \cdot (\mathbf{1} - \boldsymbol{w}) = 0$ where $\boldsymbol{w} = x_1 \boldsymbol{s}_1 + \cdots + x_\mu \boldsymbol{s}_\mu$. Furthermore, under these conditions, standard deviation is proportional to $\|\mathbf{1} - \boldsymbol{w}\|_2^2$.*

Proof. See Appendix A.4 in [1].

If $\boldsymbol{S}$ spans $\mathbb{R}^n$, then by definition of a generating family we can find $\{x_1, \ldots, x_\mu\}$ such that $\boldsymbol{w} = \mathbf{1}$.[4] Concretely, we can construct an orthonormal basis of $\mathbb{R}^n$ from vectors of $\boldsymbol{S}$ and project $\mathbf{1}$ onto each coordinate. We dub this method of computing $\boldsymbol{x}$ the *minimum variance tally-combining algorithm* (MV, Table 1). When $\boldsymbol{S}$ span $\mathbb{R}^n$, the MV algorithm gives an exact result (zero bias and variance).

Table 1. Algorithm for minimum variance tally combining (MV).

```
Input: $\boldsymbol{S} = \{\boldsymbol{s}_j\}$, $\boldsymbol{T}$, $\mu$, $n$
Output: $V_x$, $\boldsymbol{x}$, $\boldsymbol{w}$

  $Z \leftarrow \emptyset$
  For each $\boldsymbol{s}_j \in \boldsymbol{S}$, if $\boldsymbol{s}_j$ is linearly independent from $Z$, $Z \leftarrow Z \cup \boldsymbol{s}_j$
  $\widehat{Z} \leftarrow \text{GramSchmidtOrthogonalisation}(Z)$
  For each $\hat{\boldsymbol{z}}_j$, let $\hat{x}_j \leftarrow \mathbf{1} \cdot \hat{\boldsymbol{z}}_j$
  $\boldsymbol{w} \leftarrow \sum_j \hat{x}_j \cdot \hat{\boldsymbol{z}}_j$
  $M \leftarrow (\boldsymbol{z}_j \cdot \hat{\boldsymbol{z}}_\ell)_{j,\ell}$
  $\boldsymbol{x} \leftarrow (M^\top)^{-1} \cdot \boldsymbol{w}$
  $V_x \leftarrow \boldsymbol{x} \cdot \boldsymbol{T}$
  Return $V_x, \boldsymbol{x}, \boldsymbol{w}$
```

[4] The average value of μ such that $\boldsymbol{S}$ spans $\mathbb{R}^n$ is $\sum_{k=1}^{n} \frac{2^k}{2^k - 1}$. See [4] for more precise results.

However, when $\boldsymbol{S}$ does not span $\mathbb{R}^n$, the MV algorithm can only find a vector $\boldsymbol{w}$ close to $\mathbf{1}$, namely the closest such vector in terms of Euclidean distance that can be expressed in terms of vectors in $\boldsymbol{S}$. This is still the solution resulting in the smallest variance, but no longer the solution with the least bias!

This leads us to consider the following approach: we can construct tally-combining algorithms that guarantee zero bias, and select amongst these an algorithm that minimizes variance. Indeed, the constraint $\mathbf{1} \cdot (\mathbf{1} - \boldsymbol{w}) = 0$ can be guaranteed by determining x_1 as a linear function of other variables[5]. It remains to minimize $\|\mathbf{1} - \boldsymbol{w}\|_2^2$ which is simply a quadratic form in $\mu - 1$ variables. Therefore its minimum is easy to find as it amounts to solving a linear system in $\mu - 1$ rational variables. We call the corresponding algorithm the *zero-bias minimum variance tally-combining algorithm* (ZBMV, Table 2). In Table 2, "symbolic expression" refers to the notion that $x_1, \ldots, x_\mu$ are not evaluated but are symbols to be manipulated formally.

Table 2. Algorithm for zero-bias minimum variance tally combining.

```
Input: $\boldsymbol{S} = \{\boldsymbol{s}_j\}$, $\boldsymbol{T}$, $\mu$, $n$
Output: $V_x$, $\boldsymbol{x}$

  1. Let $x_1$ be the symbolic expression $\frac{1}{\mathbf{1} \cdot \boldsymbol{s}_1}\left(n - \sum_{j=2}^{\mu} x_j(\mathbf{1} \cdot \boldsymbol{s}_j)\right)$
  2. Let $D$ be the symbolic expression $\|\mathbf{1} - \sum_{j=1}^{\mu} x_j \boldsymbol{s}_j\|_2^2$
  3. $(x_2^\star, \ldots, x_\mu^\star) \leftarrow$ solutions of the linear system $\nabla D = 0$
  4. $x_1^\star \leftarrow \frac{1}{\mathbf{1} \cdot \boldsymbol{s}_1}\left(n - \sum_{j=2}^{\mu} x_j^\star(\mathbf{1} \cdot \boldsymbol{s}_j)\right)$
  5. $\boldsymbol{x} \leftarrow (x_1^\star, \ldots, x_\mu^\star)$
  6. $V_x \leftarrow \boldsymbol{x} \cdot \boldsymbol{T}$
  7. Return $V_x, \boldsymbol{x}$
```

5.1 Comparing Tally-Combining Algorithms

Let's consider a toy example to illustrate how the three discussed tally-combining algorithms compare. Throughout this section, we take $n = 4$, $M = 6$, $\mu = 3$, $k = 3$ and $\boldsymbol{s}_1 = (1, 1, 1, 0)$, $\boldsymbol{s}_2 = (1, 1, 0, 0)$, $\boldsymbol{s}_3 = (0, 1, 0, 1)$ and $\boldsymbol{T} = (1, 0, 0)$.[6] The results are summarized in Table 3.

Algorithm 1 (Zero-bias minimum variance). *We can express x_1 in terms of x_2 and x_3 to ensure zero bias:*

$$x_1 = \frac{1}{\mathbf{1} \cdot \boldsymbol{s}_1}(n - x_2(\mathbf{1} \cdot s_2) - x_3(\mathbf{1} \cdot s_3)) = \frac{1}{3}(4 - 2x_2 - 2x_3).$$

[5] There is nothing special about $\boldsymbol{s}_1$, any other vector of $\boldsymbol{S}$ can be used. Note that $\mathbf{1} \cdot \boldsymbol{s}_1 \neq 0$.

[6] Note that in this example, knowing the tallies t_1 and t_2 reveals one participant's vote. This privacy issue is addressed later in the paper.

Table 3. Comparison between tally-combining algorithms on the toy example.

Tally-combining algorithm	Bias $\mathbf{1} \cdot (\mathbf{1} - \boldsymbol{w})$	Variance $\\|\mathbf{1} - \boldsymbol{w}\\|_2^2$	Tally $\boldsymbol{x} \cdot \boldsymbol{T}$
Naïve algorithm	−2/3	4/3	2/3
ZBMV	0	5/7	6/7
MV	1/3	1/3	1

We are left to determine x_2 and x_3, which we choose to minimize the distance of $\boldsymbol{w} = x_1\boldsymbol{s}_1 + \cdots + x_3\boldsymbol{s}_3$ to $\mathbf{1}$, i.e. the quantity

$$\|\mathbf{1} - \boldsymbol{w}\|_2^2 = \sum_{i=1}^{n} (1 - w_i)^2 = (1 - x_1 - x_2)^2 + (1 - x_1 - x_2 - x_3)^2 + (1 - x_1)^2 + (1 - x_3)^2 = \frac{1}{3}(4 + 5x_2^2 + 2x_2(x_3 - 3) + 3x_3^2 - 2x_3)$$

This achieves its global minimum value of $5/7$ at $x_2^\star = 4/7$ and $x_3^\star = 1/7$. Therefore, we have: $\boldsymbol{x} = \frac{1}{7}(6, 4, 1)$. In particular, $\boldsymbol{w} = x_1^\star\boldsymbol{s}_1 + \cdots + x_3^\star\boldsymbol{s}_3 = \frac{1}{7}(10, 11, 6, 1)$ (note that computing this vector is not necessary for the algorithm).

Algorithm 2 (Minimum variance). *We begin by computing an orthonormal basis $\hat{Z}$ from $\boldsymbol{S}$: $\hat{\boldsymbol{z}}_1 = \frac{1}{\sqrt{3}}(1, 1, 0, 0)$ $\hat{\boldsymbol{z}}_2 = \left(\frac{1}{\sqrt{6}}, \frac{1}{\sqrt{6}}, -\sqrt{\frac{2}{3}}, 0\right)$ $\hat{\boldsymbol{z}}_3 = (-\frac{1}{\sqrt{6}}, \frac{1}{\sqrt{6}}, 0, \sqrt{\frac{2}{3}})$ which gives $\hat{x}_1 = \sqrt{3}$, $\hat{x}_2 = 0$, $\hat{x}_3 = \sqrt{2/3}$, from which we get $\boldsymbol{w} = \frac{1}{3}(2, 4, 3, 2)$ and finally $\boldsymbol{x} = \left(1, -\frac{1}{3}, \frac{2}{3}\right)$.*

As expected this tally-combining algorithm has smaller variance (since $\|\mathbf{1} - \boldsymbol{w}\|_2^2 = 1/3$), compared with the ZBMV algorithm in of Algorithm 1, but its bias is not guaranteed to be zero (since $\mathbf{1} \cdot (\mathbf{1} - \boldsymbol{w}) = 1/3$).

Algorithm 3 (Naïve tally combining). *Let's use the naïve tally-combining algorithm, i.e., $\boldsymbol{x} = \frac{M}{\mu k}\mathbf{1}$. We assume here that $M = 6$, $\mu = 3$ and $k = 3$ so that $\boldsymbol{x} = \frac{2}{3}\mathbf{1}$, yielding $\boldsymbol{w} = (\frac{4}{3}, 2, \frac{2}{3}, \frac{2}{3})$. The bias for this algorithm is $-2/3$, however this algorithm has larger variance than the other two, since $\|1 - \boldsymbol{w}\|_2^2 = 4/3$.*

6 Privacy of Parallel OV-Net

In this section we investigate the decrease in privacy which we can expect due to the multiple parallel elections which are tallied individually, thus giving the adversary extra information. As an example, let us consider a simple referendum. If the outcome is unanimous , we of course lose privacy. However, the probability of this might be small. However, if we split the voters into two elections, the probability is roughly the square root of the old probability, i.e. much higher.

Recall that M is the number of the parallel and independent elections, n is the total number of voters and k is the number of elections that each voter has randomly chosen to participate in. We denote by M_i the set of voters who participated in election i and we consider that the elections are enumerated from 1 to M. Let $\text{Res}(M_i)$ be the random variable that gives the number of 'Yes' votes in the set M_i. We recall also that Y_i is the random variable that gives the number of voters in the set M_i.

6.1 Definitions and Assumptions

To quantify privacy, we use the δ-privacy definition for voting from [14] which assumes that, besides the voting elements of a voting protocol, there exists an additional party called an observer O, who can observe publicly available information. Moreover, we assume that among the n honest voters, there exists a voter V_{obs} who is under observation. For the sake of clarity, V_{obs} will refer at the same time to the voter under observation and to its vote.

Definition 2 *Let P be a voting protocol and V_{obs} be the voter under observation. We say that P achieves δ-privacy if the difference between the probabilities*

$$\mathbb{P}[(\pi_O||\pi_{V_{obs}}(\mathit{Yes})||\pi_v)^{(l)} \rightarrow 1] \text{ and } \mathbb{P}[(\pi_O||\pi_{V_{obs}}(\mathit{No})||\pi_v)^{(l)} \rightarrow 1]$$

is δ-bounded as a function of the security parameter ℓ, where π_O, $\pi_{V_{obs}}$ and π_v are respectively the programs run by the observer O, the voter under observation V_{obs} and all the honest voters v (clearly without V_{obs}).

To calculate the privacy we use the following result from [14]

$$\delta(n) = \sum_{r \in M^*_{\text{Yes,No}}} (A_r^{\text{No}} - A_r^{\text{Yes}}) \tag{1}$$

where $M^*_{\text{Yes,No}} = \{r \in \mathbb{R} : A_r^{\text{Yes}} \leq A_r^{\text{No}}\}$, $\mathbb{R}$ is the set of all possible election results and A_r^j denotes the probability that the choices of the honest voters yield the result r of the election given that V_{obs}'s choice is j.

We consider a referendum with n honest voters with a uniform distribution between yes and no votes. For simplicity, we will assume that nobody abstains. We also assume that no voters are corrupted. This is reasonable, since instructing corrupted voters to vote in a special way does not give further advantage compared to simply knowing the corrupted voters' votes. Moreover, we assume that at least one of the elections in which V_{obs} participated is surviving.

6.2 Basic Cases: $M = k = 1$ and $M \geq 1, k = 1$

The δ for a single referendum is:

$$\begin{aligned}\delta(n) &= \left(\frac{1}{2}\right)^n \frac{1}{n} \sum_{a=0}^{n} \binom{n}{a} |2a - n| \\ &= \begin{cases} 2^{-n}\binom{n}{\frac{n}{2}} & \text{if } n \text{ is even} \\ \dfrac{2^{1-n}}{n}\binom{n}{1+[\frac{n}{2}]}\left(1+\left[\dfrac{n}{2}\right]\right) & \text{Otherwise} \end{cases}\end{aligned}$$

where the first equality holds using the result from (1) and the second one using the binomial theorem.

The formula above refers to the case $M = k = 1$ where all voters had chosen to vote in the same and unique election 1. For the case $M > 1$ and $k = 1$, δ becomes a random variable and the expected value of δ of the election in which V_{obs} is participating can be defined as follows:

$$\delta_{\text{expected}}(n, M) = \sum_{n'=1}^{n} \mathbb{P}(Y_i' = n')\delta(n') \tag{2}$$

where Y_i' is the random variable that gives the number of voters who participated in the election i, including V_{obs}; and $Y_i' \sim 1 + \text{BD}(n-1, \frac{k}{M})$. Equation (2) for $k = 1$ and $M > 1$ becomes:

$$\delta_{\text{expected}}(n, M) = \sum_{n'=1}^{n} \binom{n-1}{n'-1} \left(\frac{1}{M}\right)^{n'-1} \left(1 - \frac{1}{M}\right)^{n-n'} \delta(n')$$

Figure 2 shows that privacy is almost lost when $M \gg n$.

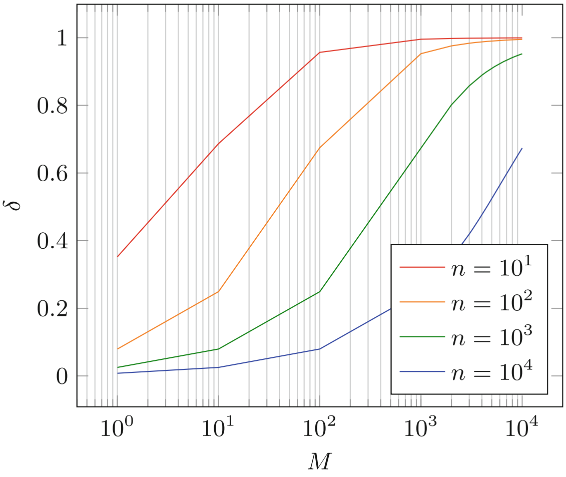


Fig. 2. The relationship between M and δ_{expected} for different values of $n = 10, 10^2, 10^3, 10^4$.

6.3 General Case

In this part we give a general formula of δ. To this end, we consider the following. Let $y = (y_1, \ldots, y_M)$ be an assignment of voters such that $\text{Card}(M_i) = y_i$ for $i \in [1, M]$. We can obtain all the possible assignments of voters by respecting the condition $\sum_{i=1}^{M} y_i = nk$. Let $r = (r_1, \cdots, r_M)$ be a possible result corresponding to the assignment y with $r_i = \mathrm{Res}(M_i)$ for $i \in [1, M]$. r verifies the conditions $(\sum_{i=1}^{M} r_i) \bmod k = 0$ and $r_i \leq y_i$ for $i \in [1, M]$. Remember that $\mathrm{Res}(M_i)$ gives the number of "Yes" votes in M_i. We have $\mathrm{Res}(M_i) \sim \mathrm{BD}(y_i, \frac{1}{2})$ for $i \in [1, M]$. Intuitively, δ can be expressed as the following:

$$\delta(n, M, k) = \sum_{y_1 + \cdots + y_M = nk} \mathbb{P}(Y_1 = y_1, \ldots, Y_M = y_M) \cdot \sum_{r \in M^*_{\mathrm{Yes,No}}} (A_r^{\mathrm{No}} - A_r^{\mathrm{Yes}})$$

By definition of A_r^j we have $A_r^j = \mathbb{P}(\mathrm{Res}(M_1) = r_1, \ldots, \mathrm{Res}(M_M) = r_M / V_{\mathrm{obs}} = j)$ with $j \in \{\mathrm{Yes, No}\}$.

To proceed we will introduce an additional notation. Remember that M_i denotes the voters in election i. Define Σ_k as the subsets of $\{1, \ldots, M\}$ of cardinality k. For $\sigma \in \Sigma_k$ we define $M'_\sigma = \bigcap_{i \in \sigma} M_i$, i.e. the voters participating in the elections in the set σ. Note that the assignment of voters to elections is uniformly random, i.e. each voter is assigned uniformly and uniquely to a M'_σ. Also Z_σ is the random variable determining the number of voters in M'_σ.

There are $c = \binom{M}{k}$ possible M'_σs . Suppose that σs are enumerated from 1 to c. Let $z = (z_{\sigma_1}, \ldots, z_{\sigma_c})$ be an assignment of voters such that $z_{\sigma_i} = \mathrm{Card}(M'_{\sigma_i})$, for $(\sigma_i, i) \in \Sigma_k \times [1, c]$. All the possible assignments of voters z are obtained by respecting the condition $\sum_{\sigma_i \in \Sigma_k} z_{\sigma_i} = n$.

The variables Z_σ, $\sigma \in \Sigma_k$ correspond to the problem of putting n indistinguishable balls into c distinguishable boxes, i.e. the vector $Z = (Z_{\sigma_1}, \ldots, Z_{\sigma_c})$ follows a multinomial distribution with equal parameters $p_i = 1/c$, and $\sum_{\sigma \in \Sigma} z_\sigma = n$ including V_{obs}. We can now calculate the probability for the assignment of the voters, and rewrite our formula as:

$$\delta(n, M, k) = \sum_{z_1 + \cdots + z_c = n} \mathbb{P}(Z_{\sigma_1} = z_{\sigma_1}, \cdots, Z_{\sigma_c} = z_{\sigma_c}) \cdot \sum_{r \in M^*_{\mathrm{Yes,No}}} (A_r^{\mathrm{No}} - A_r^{\mathrm{Yes}})$$

Let $r' = (r'_{\sigma_1}, \ldots, r'_{\sigma_c})$ such that $r'_{\sigma_i} = \mathrm{Res}(M'_{\sigma_i})$ for $(\sigma_i, i) \in \Sigma_k \times [1, c]$. The variables $\mathrm{Res}(M'_\sigma)$, $\sigma \in \Sigma_k$, are independent and follow the binomial distribution of parameters z_σ and $1/2$.

In the case $M = c$, which means $k = M - 1$ or $k = 1$, there is a one-to-one correspondence between the sets $(M_i)_{i \in [1,M]}$ and $(M'_\sigma)_{\sigma \in \Sigma_k}$. However this is not true in general and we have a relation between r and r' defined by the function f as follows:
$\begin{pmatrix} r_1 \\ \vdots \\ r_M \end{pmatrix} = B \cdot \begin{pmatrix} r'_{\sigma_1} \\ \vdots \\ r'_{\sigma_c} \end{pmatrix} = f(r'_{\sigma_1}, \cdots, r'_{\sigma_c})$ where $B = (b_{i\sigma})_{\substack{1 \leq i \leq M \\ \sigma \in \Sigma_k}}$ and $b_{i\sigma} = \mathbf{1}_{i \in \sigma}$.

We can now calculate the probability A_r^{v} as: $A_r^{\mathrm{v}} = \sum_{r' | r = f(r')} A'^{\mathrm{v}}_{r'}$ and we have: $A'^{\mathrm{v}}_{r'} = \mathbb{P}(\mathrm{Res}(M'_{\sigma_1}) = r'_{\sigma_1}, \cdots, \mathrm{Res}(M'_{\sigma_c}) = r'_{\sigma_c} / V_{\mathrm{obs}} = \mathrm{v})$.

Suppose that V_{obs} is in the subset M'_{σ_1}. It is symmetric to choose any other subset. We have: $A'^{\text{v}}_{r'} = \left(\frac{1}{2}\right)^{z_{\sigma_1}-1} \cdot h(z_{\sigma_1}, r'_{\sigma_1}) \cdot \prod_{i=2}^{c} \left(\frac{1}{2}\right)^{z_{\sigma_i}} \cdot \binom{z_{\sigma_i}}{r'_{\sigma_i}}$ where

$$h(x,y) = \begin{cases} \binom{x-1}{y-1} \text{ if v} = \text{“Yes”} \\ \binom{x-1}{y} \text{ if v} = \text{“No”} \end{cases}$$

Remember that: $M^*_{\text{Yes,No}} = \{r' : A'^{\text{Yes}}_{r'} \leq A'^{\text{No}}_{r'}\}$, and $A'^{\text{No}}_{r'} \geq A'^{\text{Yes}}_{r'}$ is true when $r'_{\sigma_1} \in [0, [\frac{z_{\sigma_1}}{2}]]$. We have $\sum_{r'_1=0}^{[\frac{z_{\sigma_1}}{2}]} (A'^{\text{No}}_{r'} - A'^{\text{Yes}}_{r'}) = \frac{1}{2} \sum_{r'_1=0}^{z_{\sigma_1}} |A'^{\text{No}}_{r'} - A'^{\text{Yes}}_{r'}|$.

Since V_{obs} is in M'_{σ_1}, the vector to consider is $Z' = (Z_{\sigma_1} - 1, Z_{\sigma_2}, \cdots, Z_{\sigma_c})$. The formula of δ becomes:

$$\delta(n,M,k) = a_n \cdot \sum_{z_{\sigma_1}=1}^{n} \frac{E(z_{\sigma_1})}{z_{\sigma_1}!} \sum_{z_{\sigma_2}=0}^{n} \cdots \sum_{z_{\sigma_c}=0}^{n} \frac{\delta_{\sum_{\sigma \in \Sigma} z_\sigma, n}}{z_{\sigma_2}! \cdots z_{\sigma_c}!} \quad = a_n \cdot \sum_{z=1}^{n} \frac{E(z)}{z!} \cdot \frac{(c-1)^{n-z}}{(n-z)!}$$

with $a_n = \frac{(n-1)!}{c^{n-1}} \cdot \left(\frac{1}{2}\right)^n$, $\quad E(z) = 2^{n-z+1} \binom{z}{[\frac{z}{2}]} \cdot \left[\frac{z}{2}\right]$ and $\delta_{i,j}$ is the Kroenecker delta function.

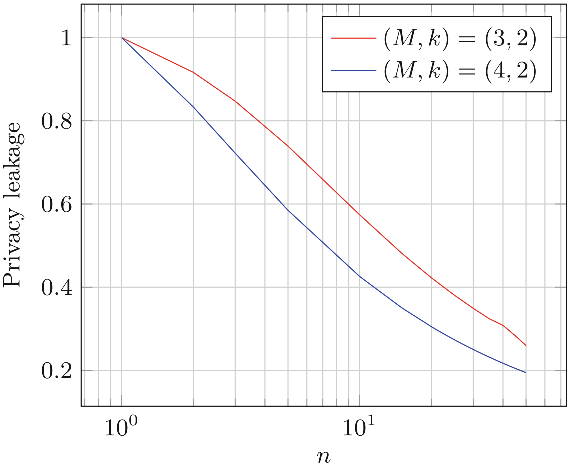


Fig. 3. Privacy leakage as function of n for the cases $(M, k) = (3, 2), (4, 2)$.

7 Conclusions and Further Research

Conclusions. In this paper, we presented a new version of the protocol OV-Net which run several elections in parallel to achieve robustness against DoS failures without having to resort to time-consuming extra rounds. We computed quantitatively the increase in robustness from having M parallel elections with each voter participating in k of these, and demonstrated that robustness can be significantly improved. The improvement in time and robustness comes at a cost in terms of accuracy and privacy. We stress that our protocol is well fitting for decision-making applications where accuracy and privacy is not of ultimate importance. We presented three different algorithms on how to optimally compute the tally using this new OV-Net version and we quantitatively measured the privacy decrease that is expected due to the multiple partial election results. The results allow the protocol initiator to choose parameters to carefully balance the wanted robustness with a controlled privacy loss, statistical loss in accuracy, as well as increased computation.

Future work. An idea to consider is redistribution i.e. elections are conducted in several electoral districts. Unlike general elections, where the final result is known for the entire country only, in redistributed elections results are consolidated per district and only then added up. This could confine problematic voters to a district of their own, as follows: partition the n voters into d districts of $n' = n/d$ voters, then run a vote in each of them. Then recompose the result by adding up the final tally. This strategy confines the DoS problem to districts that do not influence each other. However, DoS tolerance is not exactly multiplied by d because each district is not allowed to exceed k unresponsive voters. In other words, tolerance is multiplied by d as long as the constraint that there are no more than k unresponsive voters per district is respected.

Acknowledgements. PYAR acknowledge support from the Luxembourg National Research Fund (FNR) under the CORE project EquiVox (C19/IS/13643617/EquiVox/Ryan) and FEEO was supported by the FNR grant PRIDE15/10621687/SPsquared.

References

1. Bana, G., et al.: Time, privacy, robustness, accuracy: trade offs for the open vote network protocol. Cryptology ePrint Archive (2021)
2. Baudron, O., Fouque, P., Pointcheval, D., Stern, J., Poupard, G.: Practical multi-candidate election system. In: Kshemkalyani, A.D., Shavit, N. (eds.) Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing, PODC 2001, Newport, Rhode Island, USA, 26–29 August 2001, pp. 274–283. ACM (2001). https://doi.org/10.1145/383962.384044
3. Colbourn, C.J., Dinitz, J.H.: Handbook of Combinatorial Designs. CRC Press (2006)
4. Cooper, C., Frieze, A.M., Pegden, W.: On the rank of a random binary matrix. In: Chan, T.M. (ed.) Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, 6–9 January 2019, pp. 946–955. SIAM (2019). https://doi.org/10.1137/1.9781611975482.58
5. Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (1994). https://doi.org/10.1007/3-540-48658-5_19

6. Cramer, R., Franklin, M., Schoenmakers, B., Yung, M.: Multi-authority secret-ballot elections with linear work. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 72–83. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68339-9_7
7. Giustolisi, R., Iovino, V., Rønne, P.B.: On the possibility of non-interactive e-voting in the public-key setting. In: Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K. (eds.) FC 2016. LNCS, vol. 9604, pp. 193–208. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53357-4_13
8. Groth, J.: Efficient maximal privacy in boardroom voting and anonymous broadcast. In: Juels, A. (ed.) FC 2004. LNCS, vol. 3110, pp. 90–104. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27809-2_10
9. Hao, F.: A 2-round anonymous veto protocol. In: Christianson, B., Crispo, B., Malcolm, J.A., Roe, M. (eds.) Security Protocols 2006. LNCS, vol. 5087, pp. 212–214. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04904-0_29
10. Hao, F., Ryan, P.Y.A., Zielinski, P.: Anonymous voting by two-round public discussion. IET Inf. Secur. **4**(2), 62–67 (2010). https://doi.org/10.1049/iet-ifs.2008.0127
11. Hao, F., Zieliński, P.: A 2-round anonymous veto protocol. In: Christianson, B., Crispo, B., Malcolm, J.A., Roe, M. (eds.) Security Protocols 2006. LNCS, vol. 5087, pp. 202–211. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04904-0_28
12. Khader, D., Smyth, B., Ryan, P., Hao, F.: A fair and robust voting system by broadcast. Lecture Notes in Informatics (LNI), Proceedings-Series of the Gesellschaft fur Informatik (GI), pp. 285–299 (2012)
13. Kiayias, A., Yung, M.: Self-tallying elections and perfect ballot secrecy. In: Naccache, D., Paillier, P. (eds.) PKC 2002. LNCS, vol. 2274, pp. 141–158. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45664-3_10
14. Küsters, R., Liedtke, J., Müller, J., Rausch, D., Vogt, A.: Ordinos: a verifiable tally-hiding remote e-voting system. Tech. rep, Cryptology ePrint Archive (2020)
15. McCorry, P., Shahandashti, S.F., Hao, F.: A smart contract for boardroom voting with maximum voter privacy. In: Kiayias, A. (ed.) FC 2017. LNCS, vol. 10322, pp. 357–375. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70972-7_20
16. Schnorr, C.P.: Efficient signature generation by smart cards. J. Cryptol. **4**(3), 161–174 (1991). https://doi.org/10.1007/BF00196725
17. Seifelnasr, M., Galal, H.S., Youssef, A.M.: Scalable open-vote network on Ethereum. In: Bernhard, M., Bracciali, A., Camp, L.J., Matsuo, S., Maurushat, A., Rønne, P.B., Sala, M. (eds.) FC 2020. LNCS, vol. 12063, pp. 436–450. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-54455-3_31

Review Your Choices: When Confirmation Pages Break Ballot Secrecy in Online Elections

James Brunet, Athanasios Demetri Pananos, and Aleksander Essex(✉)

Western University, London, Canada
{jbrunet8,apananos,aessex}uwo.ca

Abstract. Online voting systems typically display a confirmation screen allowing voters to confirm their selections before casting. This paper considers whether a network-based observer can extract information about voter selections from the length of the exchanged network data.

We conducted a detailed analysis of the Simply Voting implementation, which had randomly varying lengths of exchanged data due to dynamic page content and gzip compression. We demonstrated that we could correctly guess a voter's selection with accuracy values ranging up to 100% in some instances. Even on more complex ballots, we generally could still rule out some combinations of candidates. We conducted a coordinated disclosure with the vendor and worked with them to roll out a mitigation.

To their credit, this discovery (and therefore its fix) was made possible by their willingness to provide a *publicly* accessible demo, which, as we will show, remains a rarity in the industry.

Keywords: Ballot secrecy · TLS · Privacy · Online voting

1 Introduction

Online voting is becoming an increasingly prevalent method of casting a ballot. Switzerland and Canada began the practice sub-nationally in 2003, with Estonia offering it nationally starting in 2005 [7]. Adoption has grown steadily since. Over 500,000 municipal voters in Ontario (Canada) cast an online ballot in 2018 [1]. Almost 250,000 Estonians (representing 45% of participating voters) cast a ballot online in the 2019 Parliamentary elections.[1] And over 650,000 online voters participated in the 2021 State election in New South Wales (Australia).[2] Despite this rapid growth, few countries have developed adequate legislation or standards for online voting systems. In this under-regulated environment, online voting providers largely set their own security requirements, which has led to mixed outcomes.

[1] https://valimised.ee/en/archive/statistics-about-internet-voting-estonia.

[2] https://elections.nsw.gov.au/About-us/Media-centre/News-media-releases/iVote-and-2021-NSW-Local-Government-elections.

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 36–52, 2022.
https://doi.org/10.1007/978-3-031-15911-4_3

In this paper, we examine the question of ballot secrecy from the network perspective. Although ballot secrecy is a well-established requirement of democratic elections, the online voting setting offers new opportunities for exploitation. For example, suppose a network observer such as a internet service provider, content delivery network, or data center could determine how you voted. In that case, they could selectively prevent your ballot from reaching the election server to unduly influence the outcome of the election. Worse, with growing precedent for service disruptions and outages due to inadequate bandwidth[3] on election night [1], a deliberate attack of this attack could escape detection.

Contributions. We present a novel ballot secrecy attack based on network traffic analysis of (encrypted) ballot confirmation pages. For a recent mayoral race in Canada, we demonstrate a classifier that could have correctly guessed voting intention for 84% of ballots based *only* on the byte-length of encrypted network traffic. Our results include:

- A detailed analysis of a real-world online voting system demonstrating the attack's effectiveness in spite of well-configured TLS and variable-length HTML and DOM elements.
- A coordinated disclosure with the affected vendor resulting in them rolling out a mitigation.
- An analysis of the broader industry's susceptibility to this attack and a discussion of mitigation options.

The rest of the paper is organized as follows: Sect. 2 presents background and related work. Section 3 recounts our efforts to reach out to vendors to seek demos to their voter interface. Section 4 describes the basics of the Simply Voting system. Section 5 describes our overall testing methodology, including technical details of our approach replicating Simply Voting's server functionality and collecting network data. Section 6 presents the results of a simple (single contest) attack on ballot secrecy. Section 7 extends the experiment to more complex ballot configurations. Finally, Sect. 8 describes our coordinated disclosure with Simply Voting, their mitigation strategy, and the approaches of the other (responsive) vendors.

2 Background and Related Work

Ballot secrecy in online elections has been studied in the context of active attacks, such as subverting TLS [2,8], exploiting implementation vulnerabilities [11,13], or by unacknowledged privileged access [4]. Little related-work has evidently explored passive attacks that focus on the *lengths* of exchanged messages. One of the first articulations of this risk is a requirement due to Volkamer and Krimmer [12] (emphasis added):

[3] https://zdnet.com/article/no-surprise-nsw-ivote-fails-during-local-council-elections/.

> *The e-voting system SHALL ensure neither the vote itself nor the number of chosen candidates (including an empty ballot), nor a spoilt vote (eg, by using the length of the protocol messages depending on the approach) can be deduced by reading transmitted voting protocol messages.*

Clark and Essex [3] considered the possibility of a network observer being able to differentiate a voter's selection based on the length of encrypted traffic sent to the election server by the voter's browser. They found Dominion Voting Systems encoded candidate names explicitly in the cast vote object. For example, they observed a vote for *Meghan Agosta* was sent in an (encrypted) `POST` as `{"ChoiceName":"Meghan Agosta"}`. They speculated this approach could be susceptible to network-based length attacks, but did not conduct an analysis.

More recently, Specter et al. [10] explored this question in the context of the Voatz mobile voting app. Like the Dominion example, Voatz explicitly encoded the chosen candidate's name, sending it to the server along with associated metadata in an HTTP `POST`. The authors observed a difference in the transmitted byte length of packets between a ballot cast for a candidate with a "short" name versus one with a "long" name.

However, our own experience examining online voting implementations has generally found cast ballot objects have a *fixed length*, with selections represented either as a code or ciphertext. This approach seemingly precludes length-based analysis—so we thought.

3 Research Question and Scope

Our study began with a hypothesis: Do ballot confirmation pages leak information about a voter's selections? In particular, if the page was generated at the server-side and sent to the client immediately prior to casting, the TLS record byte-length may reveal information about the selected candidate.

Testing this hypothesis required access to a real-world online voting implementation. However, we were unaware of any vendor who maintained a publicly accessible demonstration that we could examine. The sole exception we observed was Simply Voting, a Montreal-based online voting vendor. Simply Voting mostly focuses on *non-governmental* elections (schools, companies, unions, political parties, etc.), however they did run the elections of 28 cities (accounting for over 300,000 voters) in the 2018 Ontario Municipal Election [1].

3.1 Vendor Demo Access Requests

As explained in subsequent sections, we were able to confirm our hypothesis on Simply Voting's demo website. But what about the industry at large? Following our coordinated disclosure with Simply Voting, we decided to reach out to companies who had run (or were likely to run) a civic election in the near term.

We emailed each company identifying ourselves as cybersecurity researchers requesting a demonstration of the ballot casting experience. For each vendor, we recorded whether they responded to our request, whether we were granted access to a demo, whether it was vulnerable to length-based analysis, and if so, what mitigation strategy was employed. We gave each vendor 30 d to respond. The results are shown in Table 1. The observed mitigations are discussed in Sect. 8.

Table 1. Vendor responses to our demo request and associated findings.

	Responsive	Access granted	Vulnerable	Mitigation strategy
Dominion	No	No	Unknown	Unknown
Intelivote	No	No	Unknown	Unknown
Neuvote	Yes	Yes–Private	No	Client-side generation
Scytl	Yes	No[a]	Unknown	Unknown
Simply Voting	Yes	Yes–Public	Mitigated	Random-length padding
SwissPost	Yes	Yes–Private	No	Client-side generation
Voatz	No	No	Unknown	Unknown

[a] Agreed in principle, but access not granted by time of writing.

4 Description of Simply Voting's System

This section describes Simply Voting's process for casting ballots and evaluates the possibility of a length-based inference at different parts of this process.

4.1 Ballot Casting Process

Step 1: Logging In. The voter navigates to `demo.simplyvoting.com` and logs in with the given user ID and password. The user's full name is then included in the HTML of the subsequent pages they access during the session.

Step 2: Submitting Choice of Candidates. The voter is presented with a single ballot page, which contains a set of offices (e.g., Mayor and City Councillor) and candidates. The voter selects which candidates they would like to vote *for*, and presses the **Continue** button. This submits a form containing the voter's choices to the server represented as fixed-length codes.

Step 3: Confirmation. A confirmation page is sent to the voter from `demo.simplyvoting.com`. The served HTML content of this page contains the voter's name, as well as the name of the voter's choice of candidate. Note that static content, like images, stylesheets, and scripts, is served from a different domain, `static.simplyvoting.com`, with a different IP address.

Step 4: Review and Submission. The voter may choose to go back to the previous page and change their choices. If they do, they will again be presented with a confirmation page. If they are satisfied with their choices, the voter clicks the **Confirm** button, and their ballot is submitted to the server.

4.2 Potential Side-channel Attacks in the Ballot Casting Process

One opportunity for a length-based attack is when a voter's selections are sent to the sever, as was observed in the Voatz system [10]. The names of the chosen candidate names were being POSTed to the server as explicit, uncompressed text. By contrast, Simply Voting's system only POSTs fixed-length candidate IDs. For example, a vote for *Cassandra De Rolo* as Committee President is encoded in the HTTP request to the server as `ballot_579193[]=5724277`. Conversely, a vote for the opposing candidate, *Fernanda Rodriguez*, is represented by `ballot_579193[]=5724278`.

But what happens if the server returns a confirmation page containing the explicit names of the voter's selections?

The values of some of the DOM elements are unknown to a network observer, while others can be predicted or deduced (see Table 2 for the full list).

We hypothesized that the length and value of the chosen candidate's name had at least some effect on the size of the confirmation page and could leak information under certain conditions.

Table 2. Confirmation page DOM elements with varying values

Element	Example	Length	Predictable	Changes
CSRF token	`c9590a...67652`	Fixed	No	By session
Vote serial	`e600de...9683b`	Fixed	No	By session
Static resource version	`84932`	Fixed	Yes	Weekly[a]
Text time remaining	5 min and 0 s	Varies	Likely[b]	Every second
Integer time remaining	300	Varies	Likely[b]	Every second
Voter name	Taher Elgamal	Varies	Varies[c]	Every voter
Chosen candidate(s)	Linda Marlene Eales	Varies	–	By ballot

[a] Used by Simply Voting to periodically invalidate browser caches of their static resources. We sampled it every few days during the testing period.
[b] An observer could reasonably guess this by applying an offset to the time observed on their own confirmation page. However, off-by-one errors are possible: to make our approach as conservative as possible, we do not rely on knowing the time in our testing.
[c] Could plausibly be known by ISP or network administrator, see Sect. 5.3.

5 Methodology

To test our hypothesis that a voter's choice could correlate to the TLS record length of the ballot confirmation page, we needed to make a large volume of requests for confirmation pages and analyze the data transferred. Simply Voting's public demo of their service allows us to observe what data is transmitted from their servers in a realistic election setting. However, making tens of thousands of requests to their servers would place an undue burden on their resources and could trigger their network intrusion detection systems. Instead, we created our own server that replicates their confirmation page functionality. We also designed an application that could automatically make thousands of browser requests to this service and log the response for later analysis.

5.1 Testing a Length-Based Side-channel Attack

We created a testing system composed of two parts: a Client Application (to mimic a set of voters) and a Server Application (to mimic the online voting system). Each ballot "cast" in the experiments below corresponded to an actual HTTP request made over the internet between our local Client and cloud-based Sever applications.

We designed our applications to simulate an election where a voter is eligible to vote for one or more offices (e.g., Mayor, Councillor, Deputy Mayor), and may cast a vote for no more than *one* candidate for each office. A voter casts a single *ticket*, a combination of candidates selected for each office. This is a common electoral system for municipalities in Ontario. Some Ontario municipalities use at-large systems,[4] but this paper does not examine those elections.

5.2 Technical Implementation of the Client Application

We created the Client Application using Python, Selenium WebDriver, Google Chrome, and Wireshark. It was designed to make requests for confirmation pages, programmatically capture the response at the network layer, parse the TLS record length, and log the candidate choice and TLS record length to a file for statistical analysis. Our test bench is extensible and programmable: The client can decide which ballot to render by sending descriptive JSON to the server. The client can also set the flags to modify server behavior. For example, we implemented a flag that could programmatically enable/disable Simply Voting's `X-Ballot-Secrecy` header (see Sect. 8.3).

The Client Application takes the following steps while interacting with the Server Application:

Table 3. 2018 municipal ballot options in Ward Ennismore, township of Selwyn

Mayoral candidate	Council candidate
Linda Marlene Eales	Donna Ballantyne
Andy Mitchell	Brad Sinclair
Ron Black	ABSTAIN
ABSTAIN	

[4] https://guelph.ca/wp-content/uploads/Ward-councillors-or-councillors-at-large.pdf.

1. Client App is provided a list of offices and candidates (see e.g., Table 3).
2. Let o be the total number of offices and let $C_1, C_2 ... C_o$ represent the set of choices available to a voter for each respective office (including abstain). The set of all possible candidate combinations (also known as *tickets*) that could be submitted by a voter T, is $(C_1)(C_2)...(C_o)$. The Client Application generates $|T|n$ tickets, where n is the required sample size for each ticket.
3. In its main process, the client requests a ballot confirmation page from the Server Application using Google Chrome automated with Selenium WebDriver. The confirmation page contains one ticket in T. The main process of the Client Application then listens to a message queue.
4. A second process (the *listening process*) uses Wireshark's Python API[5] to continuously listen to responses from the server application. When a response is detected, it records the TLS record length and pushes its value into the message queue.
5. The Client Application's main process receives a TLS record length from the listening process in the message queue. Each observed record length (and the associated candidate) is appended to a CSV file. Steps 3 to 5 are repeated $|T|n$ times, until the test is complete.

5.3 Technical Implementation of the Server Application

Our goal was to replicate Simply Voting's confirmation page functionality as faithfully as possible. To that end, we studied Simply Voting's server stack and voting application by analyzing headers and interacting with their publicly accessible demo. We then matched this server stack as closely as possible, choosing popular and up-to-date software to fill gaps in the stack where Simply Voting's choice was unknown (e.g., the server OS).

Observing Simply Voting's Server Stack. We used several methods to learn about Simply Voting's application configuration. We performed an SSL test[6] to determine their supported and preferred encryption methods and analyzed the server headers sent to us while interacting with the demo application. We were able to determine the following relevant information about their server configuration:

- `demo.simplyvoting.com` reports its server software is Apache.
- The contents of the confirmation page are compressed via gzip.
- The confirmation page is streamed to the client with chunked transfer-encoding. However, in practice, only one chunk is transferred.[7]
- The TLS cipher suite on Windows and Linux desktops running Firefox or Chrome is `TLS_AES_256_GCM_SHA384`.[8]

[5] https://github.com/KimiNewt/pyshark/.

[6] https://www.ssllabs.com/ssltest/.

[7] We tested chunked transfer-encoding on and found it made no significant difference in the ability to distinguish different ballots in our tests.

[8] The chosen ciphersuite does not impact the feasibility of our attack. An observer can compute a separate record-length distribution for each observed ciphersuite.

Approximating Simply Voting's Server Stack. We rented a Virtual Private Server (VPS) from ChunkHost to use as our replicated voting server, connected it to a domain name, and obtained a TLS certificate from Let's Encrypt. We then deployed our Server Application with the following stack:

- **Debian 11.3 as the OS.** While we do not know what OS Simply Voting's servers use, Debian is an operating system with considerable market share in the server space, and 11.3 was the latest release at the time of writing.
- **Apache 2.4.52 as the server.** Simply Voting reported in its headers that it used Apache, and Apache 2.4 was the most recent minor version.
- **Flask/Python 3.9 as the web framework.** Simply Voting's web framework is unknown to us. For consistency with our client and analysis applications, we chose a Python-based web framework, and Flask is a mature Python web framework that met our relatively simple use case.
- **The TLS ciphersuite was forced to `TLS_AES_256_GCM_SHA384`.** This is the same as the TLS cipher suite preferred by Simply Voting on Windows and Linux desktops with major browsers.
- **Apache's HTTP response headers** were manually overridden to match to Simply Voting's.

Replicating Simply Voting's Web Application. Our Server System reimplements Simply Voting's ballot confirmation page. Upon receiving a request from the Client Application, the Server Application generates a confirmation page HTML document containing the data in Table 2, compresses it with GZIP, encrypts it with `TLS_AES_256_GCM_SHA384`, and serves it to the Client Application. Table 2 shows the elements with varying contents in the confirmation page, and our implementation substitutes appropriate values for all DOM elements with dynamic content:

- The Server Application generates random CSRF tokens and Vote Serials for each request.
- The Application assumes the Static Resource Version is fixed, as we observed it did not change for days at a time.
- The Server Application kept the voter's name static across our trials for several reasons. First, real-world municipal elections do not include the voter's name in the web session [1]. The voter's name may be present in non-civic elections (unions, student clubs, and political parties). Even in these cases, two further reasons exist for assuming the voter's name is known. First, the likely threat actors (e.g., internet service providers, family members, and cellular carriers) could plausibly associate a voter's TLS session with their identity and compute a distribution of TLS record lengths for a voter with that name. Second, to meaningfully abuse ballot secrecy vulnerabilities in many cases, it is necessary to already know the identity of the voter whose ballot is being observed.[9]

[9] In the case of a selective network outage attack, only the chosen candidate (not the voter's name) is relevant to the attacker.

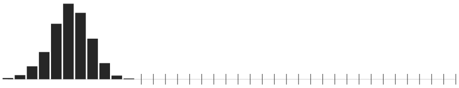
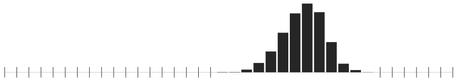
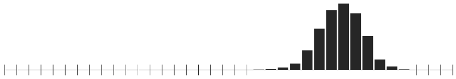
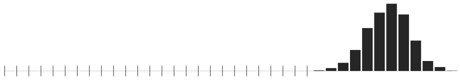
- The Application makes a conservative assumption that the time remaining varies within a 48- to 72-h window before voting closes. A more sophisticated observer may be able to increase the accuracy of their predictions by building a distribution with a more narrow time window to better approximate when a voter casts a ballot.
- The Server Application inserts the candidate choice that is requested by the Client Application.

6 Experiment 1 (Single Contest): Township of Selwyn, Ward Lakefield

6.1 Data Collection

In our first experiment, we replicated the behavior of a simple confirmation page offering a single choice for a single office, with a substantial length difference for each candidate name. One Ontario municipality that used Simply Voting during the 2018 municipal election meeting this criterion was the Township of Selwyn.[10] In 2018, voters in Ward Lakefield were eligible to vote for a Mayor, Deputy Mayor, and a Councillor. However, the positions of Deputy Mayor and Councillor were uncontested, so voters only cast a ballot for Mayor. Voters had four possible choices: Linda Marlene Eales, Andrew Mitchell, Ron Black, and Abstain.

Table 4. Observed TLS record lengths (2,000 trials per candidate)

Candidate	Frequency of occurrence	Length (Bytes)		
		Min	Mean	Max
Abstain		3,301	3,306	3,311
Ron Black		3,319	3,326	3,331
Andy Mitchell		3,322	3,329	3,334
Linda Marlene Eales		3,327	3,333	3,338

Using our Client/Server test bench described in the previous section, we cast 2,000 ballots for each candidate: While we used the actual candidate names from this contest, we simulated an equal proportion of votes for each choice instead of

[10] https://elections.amo.on.ca/web/en/municipal/19401.

the proportions of the actual election result. We recorded the TLS record length for each confirmation page returned by the Server Application. The distribution of TLS record lengths for each candidate choice is shown in Table 4.

6.2 Data Analysis

We want to estimate the probability that an encrypted vote V with byte length B is for candidate k, i.e., $\pi(V_k|B)$. To classify which candidate the encrypted vote is for a given byte length, we choose the candidate who maximizes the posterior probability:

$$\begin{aligned}\widehat{V_k} &= \underset{k \in K}{\arg\max} \left\{\pi(V_k|B)\right\} \\ &= \underset{k \in K}{\arg\max} \left\{\pi(B|V_k)\pi(V_k)\right\} .\end{aligned}$$

Generally, $\pi(B|V_k)$ is unknown. However, we can use simplifying assumptions to facilitate prediction. In particular, if we consider byte length as a categorical variable, then we can assume the likelihood for byte length is multinomial

$$\pi(B|V_k) = \text{Multinomial}(\boldsymbol{\theta}_k) .$$

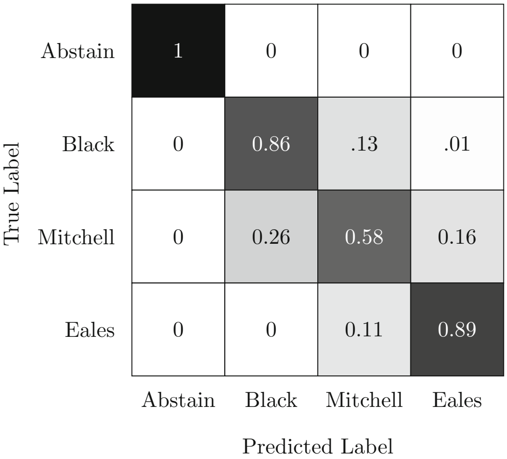
Here, the multinomial parameter $\boldsymbol{\theta}_k$ is indexed by k to allow for different candidates to have different probabilities for observing various byte lengths. Making this assumption on the likelihood leads to the *Multinomial Naive Bayes Model.* Using data with labelled votes and byte lengths, $\boldsymbol{\theta}_k$ can be estimated and then used to make predictions.

Using Python and `scikit-learn` [9], we ingest the data recorded by the Client Application and fit a Multinomial Naive Bayes Model and evaluate its out-of-sample performance on predicting which candidate a vote is for given the encrypted vote's byte length. To estimate our model's out-of-sample performance, we randomly split our data, using half to train the model and the other half to assess the accuracy of the model. The training set was used to fit our model. The performance metrics we present below are based on the predictions made on this test set. All data and code used in our analysis is available online.[11]

We evaluate model classification ability using three metrics: accuracy, precision, and recall. The ballot in this example has four choices, and we simulated an equal proportion of results for each choice. This means that the best accuracy that should be achieved for a random guess—at least in theory—is 25%.

Result. The Naive Bayes model yielded an accuracy, precision, and recall on the test set of 83%, meaning 83 of every 100 votes from a simple random sample are correctly classified using byte length alone. Class-specific accuracy varies among candidates, with some candidates seeing very high accuracy (89%) while others see smaller accuracy (58%). However, accuracy across all classes is consistently larger than the expected 25%.

[11] https://github.com/dpananos/ballot.



True Label \ Predicted Label	Abstain	Black	Mitchell	Eales
Abstain	1	0	0	0
Black	0	0.86	.13	.01
Mitchell	0	0.26	0.58	0.16
Eales	0	0	0.11	0.89

Fig. 1. Confusion Matrix (Proportions), Experiment 1. Rows normalized to sum to 1. Diagonal entries indicate class candidate-specific accuracy, while the other cells indicate proportion of votes for row candidate predicted to be the column candidate. As an example, 86% of votes for Black were correctly predicted to be for Black. 13% of votes for Black were predicted to be for Mitchell. The remaining 1% of votes for Black were predicted to be for Eales.

Figure 1, the confusion matrix, shows details about the predictions made by the Naive Bayes model on our test set. Voter choices are ordered by their mean TLS record length: It is apparent that the model is only confusing voter choices that are closest to each other in mean length. This property proves useful in later analyses of more complex elections. See *Identifying a Subset of Possible Candidate Combinations* in Sect. 7.1.

7 Additional Experiments

We conducted additional experiments with more complex confirmation pages that contain voter choices for multiple offices.

7.1 Experiment 2 (Two Contests): Township of Selwyn, Ward Ennismore

In 2018, voters in Ward Ennismore had four possible choices for mayor and three possible choices for Councillor, listed in Table 3. This results in twelve possible unique candidate combinations (tickets). We collected 500 samples per combination, for a total of 6,000 samples. Fitting a Multinomial Naive Bayes Model, we find values for accuracy, precision, and recall in Table 5. In general, performance is lower than in Experiment 1 because the length variation of different confirmation pages for the same candidate is greater. The variation increases due to

Table 5. Performance on test set by office, Experiment 2.

	Mayor			Councillor		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Naive Bayes	65%	75%	65%	50%	58%	51%
Random guessing	25%	25%	25%	33%	33%	33%

candidates for other offices being present on the confirmation page: they vary independently from the candidate being predicted.

Identifying a Subset of Possible Candidate Combinations. We also consider a more relaxed definition of violating ballot secrecy. Given a certain TLS record length, if we could identify a subset of possible candidate combinations that were chosen, that would also violate ballot secrecy. For each byte length, we counted the number of ballot configurations that produced record lengths of that byte length. Table 6 shows the proportion of ballots that have a TLS record length unique to a subset of possible candidate combinations.

Here, a possible candidate combination of n means that record length was sufficient to identify a vote to within n out of the 12 possible candidate combinations. Of note, 100% of ballots are associated with at most 11 possible candidate combinations, meaning that limited information about a voter's choice is leaked for every ballot. In other words, for all ballots, we know at least one combination of candidates that were *not* chosen by the voter.

Table 6. Proportion of ballots by possible candidate combinations, Experiment 2 (Cumulative).

	Possible candidate combinations											
	1	2	3	4	5	6	7	8	9	10	11	12
Proportion	8%	11%	14%	19%	22%	25%	37%	43%	69%	90%	100%	100%

7.2 Experiment 3 (Three Contests): Town of Ajax, Ward 1

In 2018, voters in Ajax Ward 1 had six possible choices for Mayor, three possible choices for Regional Councillor, and seven possible choices for Councillor, resulting in 126 possible candidate combinations. We collected 987–1052 samples for each combination, for a total of 128,094 samples collected. Fitting a Multinomial Naive Bayes Model, we find values for accuracy, precision, and recall in Table 7. In general, performance is lower than in Experiments 1 and 2 because of even length variations introduced by a larger set of candidates for other offices.

Candidate Combination Subsets. By viewing the TLS record lengths of different candidate combinations, we show that we can still compromise ballot secrecy (albeit to a limited extent) for all ballots in a manner similar to Experiment 2. Of the 126 possible candidate combinations (tickets), we found:

- 1% of all ballots had a unique TLS record length for that candidate combination
- 12% of all ballots cast had TLS record lengths that were shared with 10 or fewer other candidate combinations
- 53% of all ballots cast had TLS record lengths that were shared with 73 or fewer other candidate combinations
- 100% of all ballots cast had TLS record lengths that were shared with 92 or fewer other candidate combinations. In other words, for all votes cast in this election, we know at least 33 different ways to mark a ballot that was not chosen by the voter.

Table 7. Performance on test set by office, Experiment 3.

	Mayor			Councillor			Regional Councillor		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Bayes	33%	32%	33%	32%	33%	32%	63%	70%	63%
Guessing	17%	17%	17%	14%	14%	14%	33%	33%	33%

8 Mitigations

8.1 Client-Side Confirmation Page Generation

Transmitting the confirmation page over the internet can be avoided by generating the confirmation page on the client side in JavaScript. We observed the SwissPost and Neuvote systems taking this approach, rendering this particular side-channel *not-applicable.*

We met separately with representatives from Neuvote and Swiss Post and were granted private access to their (respective) demo systems. In both cases, we performed a basic analysis by casting ballots and observing the responses in Charles (an HTTP proxy) and Wireshark. We observed no ballot-related network activity in the time between selecting a candidate and rendering the confirmation page, indicating the page is generated on the client-side. We additionally observed that the cast ballot selections were encrypted at the application layer before being transmitted to the server. As expected, our experimental observations of packet lengths in Wireshark showed no perceptible correlation between candidate name length and network response length.

8.2 Fixed-Length Responses

Much discussion exists on the mitigation of length-based fingerprinting attacks, including adding padding to ensure the response is always of a fixed length. Gellert et al. describe such as scheme as "perfect length-hiding padding", but also outline major performance tradeoffs [6].

We discussed this option with Simply Voting, but the practical limitations quickly became apparent. First, the padded size would need to be larger than the largest naturally-occurring response. The second is that the gzipped length is non-linearly dependent on the content itself, requiring the padding to either be calculated and applied *after* compression or for compression to be disabled.

Padding applied dynamically as a server header after compression is an atypical use case and would likely be difficult using standard server software. Disabling compression would needlessly slow page load times, which is highly problematic for an application involving large numbers of users making requests in a short window (i.e., election night). By default, many servers only compress MIME text/HTML. One solution might be to display candidate names as fixed-length images, although this would not, on its own, rule out the possibility it could lead to other distinguishing events.

8.3 Uniformly Random-Length Padding in Response Header

Coordinated Disclosure with Simply Voting. Once we had confirmed our hypothesis with the results of Experiment 1, we contacted Simply Voting to make the coordinated disclosure. They acknowledged our result, which we discussed in-depth in a meeting. Overall, we found the interaction positive and constructive and commend them for their commitment to the disclosure process.

Following internal discussions with the engineers, they eventually settled on a mitigation involving adding a random amount of padding bytes sampled uniformly in the interval $[0, 1000)$. The sever added this padding in a new `X-Ballot-Secrecy` response header, which is now live on their ballot confirmation pages.

Analysis of Simply Voting's Fix. We implemented Simply Voting's mitigation on our cloned server. We then re-ran Experiment 1 (see Sect. 6), which had 4 ballot options. With this mitigation enabled, our prediction strategy now had an accuracy of approximately 25%—reduced to (nearly) random guessing.

However, candidates with longer names become disproportionately distinguishable in instances where the `X-Ballot-Secrecy` header sampled close to the maximal length. For example, when a voter casts a ballot for Linda Marlene Eales (the choice that produces the largest ballot selection), if the `X-Ballot-Secrecy` header is near maximal (e.g., 998, 999, or 1000 bytes), it will produce a total TLS record length that is impossible to achieve with any other candidate choice. In that case, a passive observer would be able to identify that this voter cast a ballot for Linda with a high degree of certainty.

This phenomenon also exists when the ballot secrecy header is very close to its minimal length (e.g., 0 bytes), and a voter chooses to abstain (the choice produces the shortest ballot).

To quantify this, we can perform a similar analysis to the one we did in Experiments 2 and 3; we view the maximum and minimum TLS record lengths produced by each ballot choice and identify where these distributions do not overlap. If we observe a record length outside of the distribution of one of the ballot choices, we can deduce the ballot was *not* cast for that candidate. We conducted 8,000 trials per candidate for a total sample size of 32,000. We found:

- 0.25% of all ballots had a unique TLS record length for the candidate choice
- 0.38% of all ballots had TLS record lengths that were shared with 2 or fewer other candidate choices
- 1.18% of all ballots had TLS record lengths that were shared with 3 or fewer other candidate choices
- 98.83% of all ballots had TLS record lengths within the distribution of all other candidate choices

Simply Voting's mitigation substantially lowers the risk of the attack presented in this paper. Although a practical fix under the circumstances, it still poses a risk to ballot secrecy for some voters in some cases. Client-side confirmation page generation, therefore, should remain the eventual goal.

8.4 Padding from a Gaussian Distribution

Degabriele [5] addresses the issue of overlapping uniform length distributions in the context of the CRIME/BREACH attack, where multiple observations of the same ciphertext with random padding by an attacker can be used to leak actual record lengths. The problem is similar to the limitations we identified with uniform padding in the ballot secrecy context: An attacker can observe the difference in the maximum and minimum of overlapping distributions. Degabriele proposes mitigating this by using a truncated Gaussian distribution, reducing the number of items at the tail end of the distribution. Future work should study the extent to which this approach reduces the number of clearly identifiable ballots.

8.5 Discussion and Conclusion

Using the network-observed TLS record length of the voter's vote confirmation page, our model predicted the chosen candidate in a recent real-world mayoral contest with 83% accuracy relative to random guessing (which had 25% accuracy). In more complex ballots, our model still outperformed random guessing. However, for a large subset of ballots cast in an election, we could still obtain limited information in the form of certain combinations of candidates who were *not* voted for. Validation of our models shows this performance difference is unlikely to be explained by sampling variation.

Perhaps the biggest takeaway for us, however, was how difficult it was to obtain access to voter demos. If the security of a civic election is in the public interest, companies should not need long internal deliberations to respond to a request to see what a voter already sees. In this regard, we hope the industry will eventually follow Simply Voting's example and offer demos *pro forma*.

Acknowledgements. Thanks to Simply Voting, Swiss Post, and Neuvote for providing demo access. Thanks also to Jeremy Clark, Alex Halderman, Matthew Heuman, Brian Lack, Nicole Goodman, Philip Stark, and the anonymous reviewers for their valuable feedback. This work was supported by the National Science and Engineering Research Council of Canada's Discovery Grant program.

References

1. Cardillo, A., Akinyokun, N., Essex, A.: Online voting in Ontario municipal elections: a conflict of legal principles and technology? In: Krimmer, R., et al. (eds.) E-Vote-ID 2019. LNCS, vol. 11759, pp. 67–82. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30625-0_5
2. Cardillo, A., Essex, A.: The threat of SSL/TLS stripping to online voting. In: Krimmer, R., et al. (eds.) E-Vote-ID 2018. LNCS, vol. 11143, pp. 35–50. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00419-4_3
3. Clark, J., Essex, A.: Internet voting for persons with disabilities - security assessment of vendor proposals. City of Toronto FOI Request 2014-01543 (2014). https://verifiedvoting.org/wp-content/uploads/2020/07/Canada-2014-01543-security-report.pdf
4. Culnane, C., Eldridge, M., Essex, A., Teague, V.: Trust implications of DDoS protection in online elections. In: Krimmer, R., Volkamer, M., Braun Binder, N., Kersting, N., Pereira, O., Schürmann, C. (eds.) E-Vote-ID 2017. LNCS, vol. 10615, pp. 127–145. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68687-5_8
5. Degabriele, J.P.: Hiding the lengths of encrypted messages via Gaussian padding. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, pp. 1549–1565 (2021)
6. Gellert, K., Jager, T., Lyu, L., Neuschulten, T.: On fingerprinting attacks and length-hiding encryption. In: Galbraith, S.D. (ed.) CT-RSA 2022. LNCS, vol. 13161, pp. 345–369. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-95312-6_15
7. Germann, M., Serdült, U.: Internet voting and turnout: evidence from Switzerland. Elect. Stud. **47**, 1–12 (2017)
8. Halderman, J.A., Teague, V.: The New South Wales iVote system: security failures and verification flaws in a live online election. In: Haenni, R., Koenig, R.E., Wikström, D. (eds.) VOTELID 2015. LNCS, vol. 9269, pp. 35–53. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22270-7_3
9. Pedregosa, F., Varoquaux, G., Gramfort, A., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
10. Specter, M.A., Koppel, J., Weitzner, D.: The ballot is busted before the blockchain: a security analysis of Voatz, the first internet voting application used in US. Federal elections. In: 29th USENIX Security Symposium (USENIX Security 2020), pp. 1535–1553 (2020)

11. Springall, D., et al.: Security analysis of the Estonian internet voting system. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 703–715. ACM (2014)
12. Volkamer, M., Krimmer, R.: Requirements and evaluation techniques for online-voting. In: 6th International EGOV Conference (Electronic Government), pp. 37–46 (2007)
13. Wolchok, S., Wustrow, E., Isabel, D., Halderman, J.A.: Financial cryptography, chap. Attacking the Washington, D.C. Internet Voting System, pp. 114–128 (2012)

Running the Race: A Swiss Voting Story

Thomas Haines[1](✉), Olivier Pereira[2], and Vanessa Teague[3,4]

[1] The Australian National University, Canberra, Australia
thomas.haines@anu.edu.au
[2] UCLouvain – ICTEAM – B-1348, Louvain-la-Neuve, Belgium
olivier.pereira@uclouvain.be
[3] Thinking Cybersecurity Pty Ltd., Fairfield, Australia
[4] Australian National University, Canberra, Australia
vanessa.teague@anu.edu.au

Abstract. On the 29th of March 2019 the Swiss Federal Chancellery launched a review of the procedures surrounding e-voting after numerous flaws were discovered in the Scytl-Swiss Post system sVote. On the 5th of July 2021 an independent examination of the revised Swiss Post system began, with some cantons planning to launch new trials with this system.

We summarize and reflect on our experience with the examination of the cryptographic protocol so far and muse over the future. We find that the protocol specification considerably improved over the last 3 years, both through changes in the protocol itself and through clarifications of missing elements in its specification. The clarifications also shed a new light on shortcomings of the protocol, in terms of both verifiability and privacy, including in the latest version of the system, which remains incompletely specified.

We believe that these findings illustrate virtues of the examination requirements set by the Swiss Federal Chancellery: problems can be fixed before deployment rather than being exploited by malicious parties during an election. They also illustrate the tremendous challenges of creating a secure Internet voting system, and the long road ahead.

1 Introduction

Switzerland has a long history in internet voting in political elections spanning nearly twenty years. It has also been a leader in regulating internet voting, particularly since the introduction of the Federal Chancellery Ordinance on Electronic Voting (VEleS) in 2014. This ordinance, particularly the revised version of 2018 [14], details not only security requirements for the system but requirements for the processes around the use of e-voting. Particularly crucial are the requirements which relate to transparency, for example the requirement that "Anyone is entitled to examine, modify, compile and execute the source for ideational

This paper is based on a review performed with the financial support of the Swiss Federal Chancellery.

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 53–69, 2022.
https://doi.org/10.1007/978-3-031-15911-4_4

purposes, and to write and publish studies thereon." (Art. 7b.4) We shall see that this requirement has been crucial in revealing issues in systems deployed in Switzerland.

There have historically been several different e-voting systems used by different cantons; the most prominent of these have been the CHVote open source system [3] backed by the canton of Geneva and the sVote proprietary system by Scytl and Swiss Post. Version 1.0 of the sVote protocol, which is the precursor of the current Swiss Post system [20], was used between 2016 and 2019. The system has been required since the beginning to provide individual verifiability, which it aimed to achieve through a technique called return-code voting. In the Swiss return-code voting systems, each voter receives a paper sheet containing random secret verification codes for each candidate before the election. The voter votes online by ticking their choices on web page and, in return, the browsers must show the codes that match those shown on their paper sheet. This should allow a malicious voting client to be detected should it change the voter's choices. sVote 2.1 was announced in 2018 and was designed to also provide universal verifiability. sVote 2.1 progressed through the certification process until the system was made public; at that point external experts found a large collection of errors which affected all aspects of the security of the system from privacy to verifiability. Interested readers may wish to peruse the reports by Haines et al. [4] and Locher et al. [8]. The system was withdrawn from use following these findings.

On the 26th of June 2019 the Federal Chancellery was commissioned to redesign the trial phase of e-voting with the aim to establish stable trial operations. This redesign was to have four major objectives [17]:

1. Further development of the systems
2. Effective control and oversight
3. Increasing transparency and trust
4. Closer cooperation with the academic community

The first stage of this was a dialog with various stakeholders across academia, industry and government. Based on this, the legal basis is being amended and the independent test trials have been relaunched. On the 5th of July 2021 independent experts, of which we were part, were commissioned to examine the compliance of the system with the requirements under federal law [16]. While reports from the first round of examination are available [1], the examination is still ongoing and will serve as a basis for the Federal Council's decision on whether to allow cantons to conduct e-voting. This paper summarises the situation based on Release 0.8.5.0 [18], and does not incorporate improvements made by SwissPost in their updated releases of June 24, 2022.

There is much to be applauded about how Switzerland is handling this process. However, breaking new ground is not without its difficulties. What is being attempted has never been done before and the time required to complete the process may be longer than certain stakeholders would like [13]. It is important to remember that a good certification process should *not* prematurely certify a system that does not meet requirements. It is a design feature *not* to deploy a system, even if people are expecting it to be ready, if it is not in fact ready yet.

There is no guarantee that a sufficiently secure, practical and usable system will be created in the expected timeline, if at all.

1.1 The Swiss Post Protocol

The Swiss Post e-voting system consists of numerous components which are housed either within the relevant Canton or within Swiss Post, see Fig. 1. We will now summarise the protocol, introducing the components as they become relevant by name and by the abbreviation used for the component in Fig. 1. Our protocol description is deliberately incomplete, focusing on the elements that will be useful in our further discussions. The current specifications are not particularly coherent when it comes to the components of the system. While we do our utmost to be clear, some confusion as to the participants is unavoidable in our paper since it exists in the specifications. In particular, the protocol roles include two main groups of control components, the Return Codes control components (CCR)s and the Mixing control components (CCM)s; in Fig. 1, the components denoted CC refer to a component combing the functions of a CCR and CCM. The trust assumption is that at least one member of each group of control components remains honest.

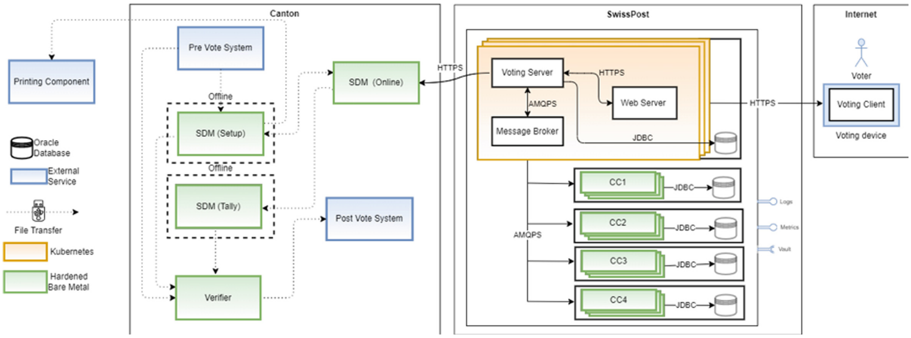


Fig. 1. System deployment - Fig. 12 from the Swiss Post E-Voting Architecture Document 1.0.0 in accordance with permitted use.

Configuration Phase. The system begins with a trusted setup component, depicted as SDM (Setup), creating the global parameters for the system. The CCMs then jointly generate a public key by running a protocol called SetupTally, at the end of which the CCMs all have a share of the secret key as does the electoral board (which does *not* appear in Fig. 1). The trusted print office together with the CCRs run a protocol called SetupVoting which generates voting cards containing that contain voter credentials and verification codes and will be sent to the voters, and stores the cryptographic information (denoted CMtable) that will be needed to recover the return codes with the help of the CCRs.

Voting Phase. The voter, having received their voting card by mail, uses the web-based voting client to create their vote. This vote is then sent via the untrusted voting server to the CCRs who, if the vote is valid, jointly compute, using the $\mathsf{CMtable}$ and the information received from the voter, the return codes to be returned to the voter. If the voter accepts their return codes, they submit the ballot casting key printed on their verification card. The CCRs check that this confirmation code is valid and if so jointly compute and return a vote cast return code.

Tally Phase. First the ballot box is cleansed to remove unconfirmed ballots and all information except the encrypted votes. The online CCMs, hosted by Swiss Post, then in sequence mix and partially decrypt the ballots.

Audit. The auditors verify the proofs generated by all the control components using $\mathsf{VerifyVotingPhase}$ and $\mathsf{VerifyOnlineTallyPhase}$, before the CCM hosted by the canton does the final mix and decryption. The auditors then check the proofs generated by the canton CCM using $\mathsf{VerifyOfflineTallyPhase}$.

In the following sections we will regularly use the same symbols as the Swiss Post specifications to facilitate interested readers making comparisons. In all cases we will first provide an explanation of the symbol in prose.

1.2 Outline

The remainder of this paper proceeds in four sections. In Sect. 2 we summarise the security required by the current draft ordinance. We have split the results of our examination of the Swiss Post system into two sections: in Sect. 3 we discuss the state of the documentation and security proofs before discussing attacks on the system in Sect. 4. Finally, we conclude in Sect. 5 and discuss possible directions that the examination process may take.

2 The Requirements

The requirements detailed in the draft ordinance are extensive and we will focus on those that pertain to the cryptographic protocol and system implementation.

Art. 3 of the April 28, 2021 draft ordinance [15] outlines four high level requirements which follow under our own headings:

Secure. "The system is designed and operated so as to guarantee verifiable, secure and trustworthy vote-casting."

Usable. "The system is easy to use for the eligible voters; account must be taken of the special needs of all voters wherever possible."

Clear. "The system and the operational procedures are designed and documented so that the details of the technical and organisational procedures can be checked and understood."

Transparent. "The general public have access to information appropriate to the addressees on how the system works and its operational processes, and there are incentives for specialists among the general public to participate."

We focus on the security and clarity: our impression is the process has the desired level of transparency and we are not qualified to assess the usability.

Requirements for complete verifiability. Complete verifiability captures the notion that it should not be possible to manipulate the result of the election without detection under certain trust assumptions on the system. To capture the latter, the ordinance considers parts of the system to be trustworthy. The notation of complete verifiability is based on two subnotations which we detail below:

Individual Verifiability. ensures that it is possible to detect manipulation of the ballot on the user's device. It should also ensure that the ballot is correctly recorded by the trustworthy part of the system. Furthermore, it should be possible for a voter who did not cast an electronic ballot to receive a proof that no ballot was received on their behalf. Individual verifiability corresponds to what is often refereed to as cast-as-intended and collected-as-cast verifiability in the literature, up to differences in the trust assumptions.

Universal Verifiability. captures that the result contains all registered votes and only those cast in conformity with the system. This corresponds to what is referred to as counted-as-collected verifiability, up to differences in the trust assumptions.

Preservation of voting secrecy. The requirements require that the secrecy of votes should be preserved provided that at least one of the control components of each group is honest and the voter's device follows the protocol and doesn't leak the vote. (This is challenging in practice, since the voter's device is expected to obtain the JavaScript code that it uses to prepare its ballot from the untrusted voting server.)

2.1 Comments on the Requirements

Positively, the draft ordinance clarifies many of the issues in the previous version. However, it continues to align more and more closely with a properly implemented version of the Swiss Post protocol. We would encourage including incentives to design stronger systems; for example by assigning grades to systems to facilitate decision making by cantons. We have heard numerous stakeholders indicate their desire to develop stronger systems after the current system meets the current requirements. But competition would be extremely hard if competition was based only on price, because stronger security does not bring an added value.

The notions of verifiability required are weaker than those common in the academic literature, which are incompatiable with a trusted print office or trusting one of the control components. In some cases, these differences allow for better usability; in others, it is unclear why the system should not be required to achieve a higher level of security. For example, return-code voting provides a tradeoff between usability and trust assumptions for which no strictly better solution is known. On the other hand, some forms of trust allowed in the setup components are unnecessary.

3 The Specification and Proofs

The Swiss Post e-voting system's protocol design is captured in the two documents entitled, "Protocol of the Swiss Post Voting System," [19] and "Swiss Post Voting System – System specification" [21]. The information about the protocol is slightly less detailed in the Protocol document than the System Specification, but the former also includes security games and proofs. We will first discuss issues with the scope of the protocol specification before discussing its alignment with the VEleS.

3.1 The Protocol Specification is too Narrow

One of the hard things in protocol design is choosing a proper layer of abstraction to describe the protocol. This abstraction should not hinder comprehension with unnecessary details but should include sufficient information to conclude the protocol is secure. The latter requirement is captured in 2.14 of the VEleS which states "One symbolic and one cryptographic proof must demonstrate that the cryptographic protocol meets the requirements in Numbers 2.1–2.12. The proofs must directly refer to the protocol description that forms the basis for system development. The proofs relating to basic cryptographic components may be provided according to generally accepted security assumptions (e.g. 'random oracle model', 'decisional Diffie-Hellman assumption', 'Fiat-Shamir heuristic')."

At the time we examined the system the following three areas were particularly noticeable as underspecified:

Authentication: The security of the system depends on how data is authenticated, which is sketched but not detailed. We pointed out the absence of specification of the authentication mechanisms, and highlighted some of the associated potential risks, which led Swiss Post to inspect these mechanisms and uncover an attack against individual verifiability.[1] This is detailed in Sect. 4.1., and we believe that this stresses the importance of the completeness of the protocol specification.

Authorisation: The security of the system also depends on when and by whom various processes can be called, which is not detailed.

Error Handling: The protocol specification focuses on protocol executions in which all the system component actions are synchronous. The verifier specification in some places specifies that verification fails in the case of inconsistency, but the verification sketch in the System Specification (for example, 12.2.3 - VerifyVotingPhase) only checks the number, not the values, of vote confirmation code attempts. In still other cases, the documents say only that inconsistencies will be investigated.

The VEleS No 2.5 requires "As a condition for the successful examination of the proof referred to in Number 2.6, all control components must have recorded the same votes as having been cast in conformity with the system. Cases where the control components show inconsistencies in this respect

[1] See https://gitlab.com/swisspost-evoting/e-voting/e-voting/-/issues/1.

must be anticipated in accordance with Number 11.11 and the procedure determined in advance." It is the last sentence of the above quote which is not completely addressed by the current documents.

Given the discrepancy that it creates w.r.t. the VEleS, the potential enormous complexity of interpreting the inconsistencies, the potential that the inconsistencies may create situations in which it is not even possible to decide whether a recorded vote matches a voter intent or not, and the likely pressure to avoid rerunning the election, we strongly recommend that this area receive far greater attention than it has to date. We have worked through the implications in some detail for the final return of vote cast confirmation codes, and made some specific recommendations in Sect. 4.2, but it may be relevant in other parts of the protocol too.

3.2 The Roles and Channels are Incompletely Aligned with the VEleS

The security model and communication channels associated to some of the protocol participants, as described in the protocol specification, seem to be incompatible with the VEleS.

In particular, the role of the auditors and of the electoral board, as described in the protocol specification, appears to be problematic.

The electoral board. The role of the electoral board is currently undefined. In Table 1 of the specification, the electoral board is not matched to any system participant of the VEleS. As such, and following Art. 2.1 of the VEleS Appendix, it should be placed within the "untrustworthy system" category. However, the protocol specification indicates, on p. 7, that "Even if some electoral board members are untrustworthy, we consider the electoral board trustworthy as a whole." We could not find any formal definition of "trustworthy as a whole".

One possible way to solve this issue would be to declare that the electoral board is an extra control component group, and therefore cannot be completely compromised. This would require extra care because the electoral board key is specified (Sect. 13.2 of the protocol specification) to be shared with Shamir's secret sharing scheme, which can accommodate any threshold, and identifying the electoral board as a control component would require it to stick to the trivial case where all key shares are necessary in order to recover the secret (because otherwise 3 out of 4 dishonest participants could collude to decrypt). And, in this case, a simpler additive secret sharing scheme can be used instead of Shamir's.

The auditors. Art. 2.2 of the VEleS Appendix forbids any outgoing communication from the auditor and from its technical aid. This is consistent with Table 2 of the protocol specification, which indicates the communication channels with the auditors and their technical aid just as in the VEleS Appendix.

The protocol specification also requires the auditors to complete VerifyOnlineTally and send information to the electoral board and last CCM before they complete the tally phase. Similarly, Fig. 23 of the specification shows that the auditors must run VerifyVotingPhase before the tally phase starts, and that the beginning of the tally phase in conditioned to a successful verification of the voting phase by the auditors.

There are many ways to address these issues. One of them would be to create an additional auditing control component group that would take the role currently assigned to the auditors in the protocol specification (the auditors in the sense of the VEleS would run the verification protocol once the election is complete). Another option would be to ask all the CCMs to run the VerifyVotingPhase themselves before they start tallying, and the electoral board to run VerifyOnlineTally before they release their keys to the offline $\texttt{CCM}_4$. We did not analyze these options in detail, and there certainly are other ones that could be considered.

4 The Bugs

Having discussed some high level issues with the protocol and requirements, we now focus on some vulnerabilities which we discovered during our examination.

4.1 Lack of Authentication: Attack on Individual Verifiability

This section of our report refers to a vulnerability disclosed to Swiss Post in March 2021,[2] prior to the current review process starting. We include it here for completeness since some of our other findings depend on this vulnerability. We also include it because the underlying vulnerability is still not patched.

When verifying signatures the Swiss Post Voting system[3] failed to check that the signatures came from the party it expected to be corresponding with. This potentially allowed attacks on integrity by spoofing the input of honest parties. These attacks could be caught by the verifier, but since the relevant parts of the verifier were not published at the point the bug was submitted (March 2021), it was not possible to verify this. Swiss Post has now confirmed how they intend to resolve this issue and, pending some slight updates to the documentation and code, the known attacks from this vulnerability should be fixed.

Key Recommendations.

Check Identity. The signature verification should check that the corresponding party is correct. This could be done by checking that the X.509 certificate's subject field contains the expected name.

Check Key Usage. All certificates in the chain should be checked to verify that they are being used for a valid purpose (using the attributes provided in RFC 5280).

Secure Initialisation. It is crucially important that the root certificates are correctly loaded. The documentation should clearly describe how this is accomplished.

[2] https://gitlab.com/swisspost-evoting/e-voting/e-voting/-/issues/1.

[3] This vulnerability was detected in version 0.7.

Details. This section of the report describes the problem as it existed in March of 2021. The current public version includes several improvements which partially address this issue; Swiss Post has confirmed they intended to update the documentation to completely address the attacks raised.

Many of the authentication checks in the system verify that the input is signed but not who it is signed by. Since the adversary has valid signing keys it can then impersonate honest parties. Examples appear to include validateChoiceCodesEncryptionKey in VotingCardSetDataGeneratorServiceImpl and validateSignature in ChoiceCodesGenerationServiceImpl.

This could allow the adversary to impersonate the one honest return code control component starting in the config phase and run undetected until the logs of the control components are examined in 12.2.3 VerifyVotingPhase.

The key issue here is that the system, when verifying signatures, does not check that the attached X.509 certificate's subject field matches the expected party or that the keys are being used for a purpose which the signer of the key's certificate intended. No check has been found which prevents the control components from impersonating the one honest control component. This would allow the one honest control component to be bypassed, which breaks cast-as-intended verification; the setup component would honestly combine the shares of the return codes but all the shares would be coming from the adversary.

No audit of the config phase described in the computational proof or system specification, at the time this issue was reported, would catch this attack on cast-as-intended. Nor was the verifier for the config phase in the repository. However, it was an open question if the attack (or a similar attack) would go undetected by the verifier specification and implementation that were (and to a significant extent are) unreleased and under development.

In conclusion, the identified vulnerability did appear to lead to manipulation that goes undetected by the voter, but not by the system, based on the then released material. However, the attack was caught by then unreleased checks.

Resolution. Swiss Post has prevented the attack detailed in this report by a manual process which checks that the certificates used in the verification are the correct certificates. This certainly prevents the specific attack detailed in this report. More details on the resolutions should appear soon when Swiss Post posts an issue on their Gitlab repo related to this finding.

Summary

At the time of writing, the underlying vulnerabilities described here are still present in the SignatureChecker class in the verifier and the various signature verification implementations in the voting system. While there are no currently known attacks which exploit the vulnerabilities, we nevertheless strongly encourage Swiss Post to patch the underlying vulnerabilities by implementing the key recommendations of this report.

Future versions of the Swiss Post Voting system aiming for higher levels of assurance may wish to dispense with certificate chains entirely and load all certificates through a manual process; this would eliminate the need to trust any root certificate authority.

4.2 Lack of Details in Handling Inconsistencies: Attack on Individual Verifiability

This section concerns the very final step of the voting phase, in which a voter enters her ballot casting key $\texttt{BCK}_{\texttt{id}}$ at her client, which transforms it into a confirmation key $\texttt{CK}_{\texttt{id}}$ and sends it to the voting server. She should receive the correct Vote Cast Return Code $\texttt{VCC}_{\texttt{id}}$ only if her ballot will be included. For reasons of space our description here is necessarily incomplete—more information can be found in our report from Round 1 of the examination [5].

The adversary's objective is either to return the correct $\texttt{VCC}_{\texttt{id}}$ to the voter, while producing a vote transcript that leads to the rejection of her vote, or to produce a vote transcript that leads to the inclusion of a vote for which the voter never entered her ballot casting key $\texttt{BCK}_{\texttt{id}}$.

The attacks described in this report rely on some inconsistencies between the logs of different CCRs for the vote confirmation phase.[4] We find it fairly difficult to understand how the system would behave, should those inconsistencies happen. We believe that the treatment of these inconsistencies should be an explicit part of the protocol specification, and that the security proof should demonstrate why this treatment is compatible with the FCh VEleS.

Our analysis focuses on specific examples. We do not currently have a proof that the proposed modifications in the protocol are sufficient, because there may be other attacks along similar lines.

What Inconsistent Logs Should be Permitted? Let us consider CCR logs that are almost, but not perfectly, consistent. This may be due to communication mishaps, a corrupted voting server, or one or more malicious CCRs.

We focus on the confirmation logs ($\texttt{L}_{\texttt{confirmed},j}$) and, in the rest of this discussion, we omit $\texttt{lVCC}_{\texttt{id}}$ and the ZKPs, because we assume these are honestly generated, consistent with the other data, and pass verification.

Omission. Suppose three CCRs show a certain confirmation attempt but one missed it, so their logs look like: (where $\texttt{vc}_{\texttt{id}}$ is the verification card identifier)

$$\begin{aligned}&\texttt{CCR}_j : (\texttt{vc}_{\texttt{id}}, 1, \texttt{CK}_{\texttt{id}}, *, *) \text{ for } j = 1, 2, 3.\\ &\texttt{CCR}_4 : \text{No record for } \texttt{vc}_{\texttt{id}}\end{aligned}$$

Such logs could appear in a scenario like the following one, in which a dishonest $\texttt{CCR}_4$ colludes with a dishonest voting client and Voting Server (VS).

1. The client and server-side components all perform the vote-sending and Choice Return Code generation and return honestly. The client displays the (correct) Choice Return Code to the voter.

[4] This was reported to SwissPost as a gitlab issue which is currently private.

2. The voter enters his true Ballot Casting key $\texttt{BCK}_{\texttt{id}}$. The client honestly computes $\texttt{CK}_{\texttt{id}}$ and sends it to the Voting Server.
3. The Voting Server honestly forwards $\texttt{CK}_{\texttt{id}}$ to all the CCRs.
4. The honest CCRs ($j = 1, 2, 3$) perform all the steps of Sect. 12.2.2.2 of the protocol specification correctly, including logging, and return long vote cast return code $\texttt{lVCC}_{\texttt{id},j}$ ($j = 1, 2, 3$) to the Voting Server.
5. Cheating $\texttt{CCR}_4$ computes $\texttt{lVCC}_{id,4}$ correctly, *but logs nothing and returns the value secretly to the Voting Server.*
6. The Voting server makes whatever logs are specified when it receives only three responses ($j = 1, 2, 3$). (This is currently not explicitly specified in Sect. 12.2.2.3.)
7. *The Voting server also computes correctly (but does not log) the value of* $\texttt{lVCC}_{id}$ *derived from a correct execution of 12.2.2.3 using the* $\texttt{lVCC}_{id,j}$*'s received from honest CCRs (*$j = 1, 2, 3$*), plus the* $\texttt{lVCC}_{id,4}$ *it received out-of-band from the cheating* $\texttt{CCR}_4$. This result should correspond exactly to an honest execution with a valid Vote Cast Return Code, and should therefore find a match in the CMTable at Step 3 of Sect. 12.2.2.3.
8. *The Voting Server then sends the (correct)* $\texttt{VCC}_{\texttt{id}}$ *value back to the colluding voting client out-of-band.*

Thus the voter submitted his $\texttt{BCK}_{\texttt{id}}$ and received a final confirmation with the correct code.

However, such logs could also appear in a scenario like the following, in which a dishonest $\texttt{CCR}_4$ colludes with a dishonest voting client, while the VS is honest.

1. *The client modifies the vote choices made by the voter and submits an incorrect ballot to the Voting Server. The CCRs compute the corresponding choice return codes, which the voter rejects since they do not match her choices.*
2. The voter does not enter her Ballot Casting key $\texttt{BCK}_{\texttt{id}}$. *The client guesses a* $\texttt{BCK}_{\texttt{id}}$ *value, computes the corresponding* $\texttt{CK}_{\texttt{id}}$ *and sends it to the Voting Server.*
3. The Voting Server honestly forwards $\texttt{CK}_{\texttt{id}}$ to all the CCRs.
4. The honest CCRs ($j = 1, 2, 3$) perform all the steps of Sect. 12.2.2.2 of the protocol specification correctly, including logging, and return $\texttt{lVCC}_{\texttt{id},j}$ ($j = 1, 2, 3$) to the Voting Server.
5. *Cheating* $\texttt{CCR}_4$ *does nothing, and returns no value to the Voting server.*
6. The Voting server makes whatever logs are specified when it receives only three responses ($j = 1, 2, 3$). (This is currently not explicitly specified in Sect. 12.2.2.3.) It also returns no Vote Cast Return code to the voter.

Thus the voter never entered her $\texttt{BCK}_{\texttt{id}}$ and received no Vote Cast Return code. (These logs could of course also be the result of other scenarios – we are just describing two examples that result from opposite voter actions and views.)

Message Reordering. Now suppose the CCR logs show the same (two) confirmation attempts, but in a different order, so their logs look like:

$$\mathtt{CCR}_j : (\mathtt{vc_{id}}, 1, \mathtt{CK_{id}}, *, *), (\mathtt{vc_{id}}, 2, \mathtt{CK2_{id}}, *, *) \text{ for } j = 1, 2, 3.$$
$$\mathtt{CCR}_4 : (\mathtt{vc_{id}}, 1, \mathtt{CK2_{id}}, *, *), (\mathtt{vc_{id}}, 2, \mathtt{CK_{id}}, *, *)$$

These logs could be the result of various scenarios very similar to the previous ones. For instance, it may be that the voting client was honest, the voter entered a correct $\mathtt{BCK_{id}}$ value, and a correct $\mathtt{CK_{id}}$ was sent to the voting server, but the malicious voting server created $\mathtt{CK2_{id}}$ as well and sent the values $\mathtt{CK_{id}}$ and $\mathtt{CK2_{id}}$ to the first three CCRs, and the values $\mathtt{CK2_{id}}$ and $\mathtt{CK_{id}}$ to $\mathtt{CCR}_4$. The corrupted voting server may then decide to send the correct Vote Cast Return code to the voting client, after reordering the responses from $\mathtt{CCR}_4$. The voter would then have a complete voting session. In another scenario, the voting server would not send the correct Vote Cast Return code to the voter. In yet another scenario, the voting client is corrupted, and both $\mathtt{CK_{id}}$ and $\mathtt{CK2_{id}}$ are incorrect values.

Divergence. Now suppose all the CCRs show two confirmation attempts, but all with different values, so their logs look like:

$$\mathtt{CCR}_1 : (\mathtt{vc_{id}}, 1, \mathtt{CK1_{id}}, *, *), (\mathtt{vc_{id}}, 2, \mathtt{CK2_{id}}, *, *)$$
$$\mathtt{CCR}_2 : (\mathtt{vc_{id}}, 1, \mathtt{CK3_{id}}, *, *), (\mathtt{vc_{id}}, 2, \mathtt{CK4_{id}}, *, *)$$
$$\mathtt{CCR}_3 : (\mathtt{vc_{id}}, 1, \mathtt{CK5_{id}}, *, *), (\mathtt{vc_{id}}, 2, \mathtt{CK6_{id}}, *, *)$$
$$\mathtt{CCR}_4 : (\mathtt{vc_{id}}, 1, \mathtt{CK7_{id}}, *, *), (\mathtt{vc_{id}}, 2, \mathtt{CK8_{id}}, *, *)$$

These logs could be the result of a malicious voting server who sent random $\mathtt{CK_{id}}$ values to the CCRs – and this could happen whether or not the voter entered his correct $\mathtt{BCK_{id}}$. Alternatively, they could be the result of an honest voter entering his correct $\mathtt{BCK_{id}}$ on a second attempt, resulting in the submission of $\mathtt{CK1_{id}}$ and $\mathtt{CK2_{id}}$ to all the CCRs, and then of incorrect behavior by $\mathtt{CCR}_2$, $\mathtt{CCR}_3$ and $\mathtt{CCR}_4$, which would log random $\mathtt{CK_{id}}$ values and may or may not compute and return the correct $\mathtt{lVCC}$ codes to the voting server.

Discussion. In all three cases, there is no appropriate consistent information from any single attempt to extract a valid Vote Cast Code. Also, it is not possible to decide, just from these logs, what went wrong: these transcripts could be the result of an innocent communication problem, of a corrupted VS, or of the corruption of one or more CCRs.

In the message reordering case, the logs offer sufficient information to verify whether the correct $\mathtt{CK_{id}}$ value is in the list, based on the $\mathtt{lVCC_{id}}$ values from the logs and on the $\mathsf{CMtable}$. In the other two cases, the logs offer no way to decide whether the correct $\mathtt{CK_{id}}$ is in the list.

It is also unclear whether a $\mathtt{VCC_{id}}$ would be returned to the voter in any of these cases.

What do the Specification Documents Say About These Cases? We inspect the different available documents in order to try to interpret what would happen.

Protocol Specification. The scenarios above describe some inconsistencies between the logs of different CCRs for the vote confirmation phase. At present, in version 0.9.11 of the protocol specification documents, the consistency checks described in the $\mathsf{VerifyVotingPhase}$ algorithm (Sec. 12.2.3), which decide whether votes are tallied, are only incompletely specified—it is not clear whether the proposed scenarios would pass or not.

Step 5 of the verification of the CCR logs indicates: "Check the equality of $\mathtt{vc}_{\mathtt{id}}$ and confirmation attempts number in $\{\mathsf{L}_{\mathtt{confirmed}_j}\}_{j=1}^{m}$. Our understanding is that the "Omission" case would fail on this criterion, but that the "Message reordering" and the "Divergence" cases would pass, since all the CCRs have 2 attempts for $\mathtt{vc}_{\mathtt{id}}$.

The presence of extractable short Vote Cast Return Codes is also verified. Here, we expect that the "Divergence" case would fail because of the absence of $\mathtt{lVCC}_{\mathtt{id},j}$ tuples in the CCR logs that make it possible to extract a return code from $\mathsf{CMtable}$. The case of the "Message reordering" is less clear: VS could have marked the ballot as extractable, and the right $\mathtt{lVCC}_{\mathtt{id},j}$ values will be found in the CCR logs, even though they won't correspond to the same attempt: even though we do not find any suggestion that an honest VS would try to reorder values coming from the CCR in order to see if they lead to an extractable code (and hence would mark the ballot as non-extractable), the VS is not trusted to follow the protocol specification and could mark the ballot as extractable. Besides, the verification process does not seem to require that the right $\mathtt{lVCC}_{\mathtt{id},j}$ values must come from identical attempt numbers in the CCR logs: this could make this ballot pass verification.

Protocol Specification, again. Much later, in Sect. 16.2 of the protocol specification document, there is an indication that auditors who find an inconsistency could start interacting with other system components, perform an analysis, which could result in a modification of the voting server and the control components' state and in the list of ballots to be included in the tally.

How are these questions handled in the security proof? The relevant section is in 16.2, where Theorem 3 formalises the idea that a voter should not receive a valid Vote Cast Return code for a vote that is not included.

The security proof does not properly cover cases like this—see [5] for details.

Verifier Specification. The verifier specification (version 0.9.1) is more demanding, and it appears from Sect. 4.1 that none of the inconsistencies that we propose would pass verification: verification step 2.43 requires strict equality accross control components of the $\mathtt{hCK}_{\mathtt{id}}, \mathtt{attempts}_{\mathtt{id}}, \mathtt{vc}_{\mathtt{id}}$ values. This would in particular imply that the "Message reordering" case, which may have passed the previous verification steps, would still result in a verification failure.

Contrary to what appears in Sect. 16.2 of the protocol specification document, the verifier specification just concludes with a failure, and there is no suggestion that any log reconciliation attempt should be made.

4.3 Lack of ZK Proofs of Correct Key Generation: Attack on Privacy

The CCMs do not prove knowledge of the secret keys corresponding to the public key that they publish. This is important since the absence of these proofs means that a minority of parties may know the secret key, which should have been generated in a distributed manner.

The following attack illustrates discrepancies between the VEleS, the protocol specification and the security proofs. Although we do not think it would work in the security model of the protocol specification, the proof does not characterise the possible attacks sufficiently. Even more importantly, this scenario shows a point in which the trust model of the protocol specification is inconsistent with the VEleS.

An attack scenario on privacy. Let us consider the following variation on the classical attack described in Sec. 13.6 of the protocol specification. We consider a case where the voting server, the election board and one of the online CCMs are controlled by the adversary. The adversary sees the inputs of the honest CCMs' public key shares $(EL_{pk,1}, EL_{pk,2})$ through the voting server (Fig. 20 of protocol specification) and creates a share which cancels them out. This is done by inverting their shares and adding one of its own $EL_{pk,3} = \frac{EL'_{pk,3}}{\prod_{i=1}^{2} EL_{pk,i}}$. The setup component acts honestly and computes $EL_{pk} = \prod_{i=1}^{2} EL_{pk,i} \cdot EB_{pk}$ which simplifies to $EL'_{pk,3} \cdot EB_{pk}$. At this point the adversary knows the secret key used to encrypt votes and can break privacy as the votes are submitted.

We observe that this attack scenario does not exist in the more abstract model that is used in the security proof, since that model considers one single online CCM (merging $\mathtt{CCM}_1$, $\mathtt{CCM}_2$ and $\mathtt{CCM}_3$).

This attack would also not work in the security model of the protocol specification, because:

1. It is considered that some electoral board members cannot be corrupted (Table 1).
2. It is considered that the auditors, among which one of them is supposed to be honest, authorize the electoral board member to reveal their secret key to the offline CCM, and this would only happen after a successful mixing, which $\mathtt{CCM}_3$ would not be able to complete. So, $\mathtt{CCM}_4$ would never receive the decryption key shares.

5 Conclusion

The Swiss regulations and processes for e-voting are world leading and we strongly advocate adoption of similar processes in countries like Australia, Estonia, and any other jurisdiction using Internet voting for political elections. The Swiss Post e-voting system is continuing to improve, gradually fixing issues it inherited from sVote. However, the system is still not complete and significant security issues are still being discovered.

This experience may feel frustrating for the stakeholders who are looking forward to a swift return of e-voting in Switzerland, especially when e-voting has been used for years.

Our feeling is rather that the process illustrates difficulties that were always there.

– The design of an Internet voting system that would offer security in a context that is suitable for government elections is widely regarded as an open question by the academic community [9,11,22].
– The other countries that decided to open their Internet voting system to public scrutiny (and many that didn't) also faced the discovery of significant security issues – see the cases of Norway, Estonia, Australia and Russia for instance [2,6,7,10,12,23].

Switzerland adopted regulations regarding the review of its Internet voting system that are well aligned with the practices adopted for other high-impact cryptographic protocols. The process is however made quite challenging because of the unique set of requirements adopted by Switzerland on the one hand, and because of the almost complete absence of existing standards regarding e-voting protocols, and on which a Swiss system could rely. As a result, we encourage all stakeholders to allow sufficient time for the system to be properly developed and reviewed before deployment. Remember that *not* certifying a non-compliant system is a desirable goal of a good process.

References

1. Federal Chancellery. E-voting: Results of the first independent examination available (2022). https://www.bk.admin.ch/bk/en/home/dokumentation/medienmitteilungen.msg-id-88085.html
2. Gaudry, P., Golovnev, A.: Breaking the encryption scheme of the Moscow internet voting system. In: Bonneau, J., Heninger, N. (eds.) FC 2020. LNCS, vol. 12059, pp. 32–49. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-51280-4_3
3. Haenni, R., Koenig, R.E., Locher, P., Dubuis, E.: CHVote system specification. Cryptology ePrint Archive, Report 2017/325 (2017). https://ia.cr/2017/325
4. Haines, T., Lewis, S.J., Pereira, O., Teague, V.: How not to prove your election outcome. In: IEEE Symposium on Security and Privacy, pp. 644–660. IEEE (2020)
5. Haines, T., Pereira, O., Teague, V.: Report on the Swiss post e-voting system (2022). https://www.newsd.admin.ch/newsd/message/attachments/71147.pdf
6. Specter, M., Halderman, J.A.: Security analysis of the democracy live online voting system (2020)
7. Halderman, J.A., Teague, V.: The New South Wales iVote system: security failures and verification flaws in a live online election. In: Haenni, R., Koenig, R.E., Wikström, D. (eds.) VOTELID 2015. LNCS, vol. 9269, pp. 35–53. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22270-7_3
8. Locher, P., Haenni, R., Koenig, R.E.: Analysis of the cryptographic implementation of the Swiss post voting protocol (2019). https://www.bk.admin.ch/bk/en/home/politische-rechte/e-voting.html

9. National Academies of Sciences, Engineering, and Medicine: Securing the Vote: Protecting American Democracy. The National Academies Press, Washington, DC (2018)
10. Pereira, O.: Individual verifiability and revoting in the Estonian internet voting system. In: Proceedings of the 7th Workshop on Advances in Secure Electronic Voting (2022). https://ia.cr/2021/1098
11. Pilet, J.-B., Preneel, B., Erzeel, S., Pereira, O.: Étude sur la possibilité d'introduire le vote internet en belgique (2020). https://elections.fgov.be/informations-generales/etude-sur-la-possibilite-dintroduire-le-vote-internet-en-belgique
12. Springall, D., et al.: Security analysis of the Estonian internet voting system. In: Proceedings of the 21st ACM Conference on Computer and Communications Security. ACM (2014)
13. Swiss Community: Swiss post e-voting to operate from 2023 (2022). https://www.swisscommunity.org/es/news-media/swisscommunity-news/swiss-post-e-voting-to-operate-from-2023
14. Swiss Federal Chancellery: Federal chancellery ordinance on electronic voting (2018). https://www.fedlex.admin.ch/eli/cc/2013/859/en
15. Swiss Federal Chancellery: Federal chancellery ordinance on electronic voting (draft of 28 April 2021) (2021). https://www.bk.admin.ch/bk/en/home/politische-rechte/e-voting.html
16. Swiss Federal Chancellery: Federal government launches examination of new e-voting system (2021). https://www.bk.admin.ch/bk/en/home/dokumentation/medienmitteilungen.msg-id-84337.html
17. Swiss Federal Chancellery: E-voting (2022). https://www.bk.admin.ch/bk/en/home/politische-rechte/e-voting.html
18. Swiss Post: E-voting documentation 0.8.5.0 (2021). https://gitlab.com/swisspost-evoting/e-voting/e-voting-documentation/-/tree/documentation-0.8.5.0
19. Swiss Post: Protocol of the swiss post voting system - version 0.9.11 (2021). https://gitlab.com/swisspost-evoting/e-voting/e-voting-documentation/-/tree/documentation-0.8.5.0/Protocol
20. Swiss Post: Swiss post voting system (2021). https://evoting-community.post.ch/
21. Swiss Post: Swiss post voting system - system specification - version 0.9.7 (2021). https://gitlab.com/swisspost-evoting/e-voting/e-voting-documentation/-/tree/documentation-0.8.5.0/System
22. U.S. Vote Foundation: The future of voting: end-to-end verifiable internet voting - specification and feasibility study (2015). https://www.usvotefoundation.org/E2E-VIV
23. Espen Zachariassen: Feil i krypteringen av e-stemmer (2013). https://www.tu.no/artikler/feil-i-krypteringen-av-e-stemmer/234436

The Effect of Exogenous Shocks on the Administration of Online Voting: Evidence from Ontario, Canada

Helen A. Hayes[1], Nicole Goodman[2](✉), R. Michael McGregor[3], Zachary Spicer[4], and Scott Pruysers[5]

[1] McGill University, Montreal, QC, Canada
helen.hayes@mcgill.ca
[2] Brock University, St. Catharines, ON, Canada
nicole.goodman@brocku.ca
[3] Toronto Metropolitan University, Toronto, ON, Canada
mmcgregor@ryerson.ca
[4] York University, Toronto, ON, Canada
zspicer@yorku.ca
[5] Dalhousie University, Halifax, NS, Canada
scott.pruysers@dal.ca

Abstract. This paper examines the impact of two exogenous shocks – a 2018 technical incident that took place in Ontario, Canada, and the COVID-19 pandemic – on the administration of local elections in Ontario. Drawing upon survey and focus group data, this paper concludes that these two exogenous shocks affected the perception and adoption of online voting on the municipal level in differential ways. We find that the COVID-19 pandemic had a greater perceived effect upon the decision to adopt online voting than the 2018 technical incident. However, the perceived effects of the 2018 technical incident were just as likely to be felt in unaffected municipalities as they were in those that had been directly affected. Municipalities that had not used online voting in 2018 and medium-sized cities were more negatively affected by the 2018 technical incident. In contrast, the perceived effects of the COVID-19 pandemic did not hinge upon the previous use of online voting, city size, or the urban/rural divide.

Keywords: Online voting · Technical incident · COVID-19 · Exogenous shocks · Policy window · Canada · Ontario

1 Introduction

Contextual circumstances can and do influence how administrators run elections. Whether reacting to economic crises [1], war [2], natural disasters [3], or public health emergencies [4], election officials around the world have had to pivot during periods of uncertainty to continue to offer regular, free, and fair elections. In particular, the COVID-19 pandemic has pushed election officials to make changes to election rules and

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 70–89, 2022.
https://doi.org/10.1007/978-3-031-15911-4_5

processes that they may not have made otherwise – including the expeditious adoption of remote voting methods such as postal and online voting [5]. The COVID-19 pandemic has also encouraged governments to streamline and modernize election processes by adopting other types of election and voting technologies. Yet, while certain events push governments to deploy election technologies sooner than they might have otherwise, the growing implementation of voting technologies brings with it the increased likelihood of technical incidents – a type of exogenous shock that may impact the ways that officials administer elections. In fact, as governments and election management bodies (EMBs) embrace voting technologies, the possibility for technical incidents increases. Recent examples of such events occurred during New South Wales's 2021 local elections in Australia [6–8], in the 2018 Pakistani general election [9], and, of relevance to this work, in Ontario's 2018 municipal elections [10].

During Ontario's 2018 municipal elections,[1] a technical issue with an online and telephone voting election service caused a voting outage in 43 municipalities across the province, forcing local election administrators to take contingency measures to ensure that all voters had a chance to participate (herein referred to as the "2018 technical incident"). Though the affected municipalities represent only a fraction of those that employed online voting in 2018, the technical incident was widely reported about in the media and caught the attention of administrators from across the province. Two years later, in 2020, Ontario was also impacted by the COVID-19 pandemic. Calls for health and safety measures altered the ways that officials carried out elections, including in the adoption and use of election technologies.

With a particular focus on online voting, this paper explores how these two shocks – the 2018 technical incident and the COVID-19 pandemic – may have: (1) impacted the administration of electronic elections in Ontario and, (2) been perceived differently across Ontario's municipalities. Ontario provides an intriguing case in which to study the impact of these shocks because of its large number of municipalities[2] that account for both urban and rural communities, small and large population sizes, and varying levels of voting technology usage. Although much research has examined the effects of the COVID-19 pandemic on election administration, this case allows us to directly compare the effects of the pandemic and a technical incident to see if and how they matter differently. This comparison, made among the same set of survey respondents, offers a novel contribution to research on the effects of unpredictable circumstances on small-scale elections reforms, including the adoption of electronic voting technologies.

We hypothesize that each shock has exerted different pressures on administrators and elections. It is likely that the 2018 technical incident increased insecurities about online voting and therefore presumably discouraged municipal uptake, while the COVID-19 pandemic encouraged local governments to adopt, or at least consider adopting, online voting. These forces are countervailing, and it is possible that both mattered, neither did, or that one mattered more than the other. We find that the COVID-19 pandemic had a greater perceived effect upon decisions of whether to adopt online voting than the 2018 technical incident, especially for those municipalities whose voting plans had been affected by the 2018 voting outage. However, these two shocks had different effects

[1] All regular municipal elections in a province occur at the same time.

[2] Ontario has 444 municipalities.

on different types of municipalities. Given the persistence of the pandemic and the increasing frequency with which technical incidents occur, our results offer important insights for both scholars and policymakers.

2 Context: Municipal Elections in Canada

Canada is a federation where municipalities are established and have their authority defined by their respective provincial governments. As such, the authority, decision-making power, and very existence of municipalities is a product of provincial legislation (i.e., they are 'creatures of the provinces'). This power extends to the regulation of municipal elections, wherein each province has one enabling piece of municipal legislation. In Ontario, the *Municipal Act* guides municipalities in most policy and legal domains. However, the *Municipal Elections Act* sets out the rules, timing, and procedures for running local elections across the province. Section 42(3) of the *Municipal Elections Act* provides for the use of alternative voting methods during municipal elections if the council of the municipality has approved the use of such methods. Likewise, Sect. 11(2) of the *Municipal Elections Act* places the responsibility for conducting the election upon the Clerk.[3]

In Ontario, there is substantial uptake of online and telephone voting among municipalities. Along with Estonia and Switzerland, Canada has one of the longest standing deployments of online voting in the world. The use of online voting in binding elections in Canada began in 2003, when 12 municipalities in Ontario adopted the technology. Since then, use of online voting has grown steadily across municipalities in Ontario.[4] In the 2018 Ontario municipal elections, 177 municipalities offered online voting, accounting for about 45% of cities and towns and 29% of the 9.4 million voters in the province [10]. In 2022, the number of municipalities offering online voting in Ontario will cross the majority threshold, with an estimated 220 out of 414 doing so.[5] In addition, many municipalities in Nova Scotia and Ontario use telephone voting as a complimentary channel, especially in communities where internet connectivity is poor or not available, or electors' digital literacy is of concern. In a majority of the municipalities employing online and/or telephone voting, paper ballots have been eliminated altogether [10].

Ontario is a unique case to study online voting development. It has a large population (comparative to other provinces and territories), a high number of municipalities (444), and the legislative framework under which it abides allows for its municipalities to make individual decisions about the voting methods they employ. In fact, since 2003, Ontario has had one of the longest standing experimentations with online voting. In contrast to most other jurisdictions where online voting adoption is implemented simultaneously,

[3] The Clerk is one of two statutory roles required for each Ontario municipality. The Clerk manages services, policy processes, elections, and matters of legislative compliance in their respective municipality.

[4] The province of Nova Scotia also uses online voting in most of its municipalities. In 2020, 39 of the 46 municipal elections held in Nova Scotia were conducted online. Although there are 48 municipalities in Nova Scotia, two municipalities had committed to using online voting, but all races were acclaimed in the 2020 election.

[5] There are 444 municipalities in Ontario. 414 of them are responsible for running local elections.

uptake in Ontario has been varied: although a growing number of municipalities have introduced the voting reform, some have switched back to paper ballots.

Despite this, Ontario has the most online voting uptake in Canada and has become a hub for electronic elections worldwide. Presently, it is the most extensive case of online voting deployment globally, and Canada is the only country wherein certain jurisdictions run online elections remotely with no paper ballot option. Given these considerations, and the fact that a technical incident occurred in the most recent Ontario municipal elections, the province provides a unique case to study the effects of exogenous shocks on the administration of local elections. Furthermore, since there is a forthcoming municipal election in October 2022, the local officials surveyed and interviewed for this paper had already made decisions about which voting modes will be used. This allowed us to learn about the first-hand considerations that factored into decision-making processes around the use of online voting for the upcoming election.

2.1 The 2018 Technical Incident

On October 22, 2018 – municipal election day in Ontario – voting websites supported by Dominion Voting Systems, one of four primary voting technology vendors in the province, slowed to an extent that it prevented voters from casting their online ballots. The slowdown occurred just before 6:00 pm EST (the polls were set to close at 8:00 pm EST) and resulted in the voting websites of 43 municipalities either not working or operating so slowly that casting a ballot was either very difficult or not possible. The company issued a press release explaining that the slowdown was the result of an unauthorized restriction in bandwidth by a third-party IT subcontractor, which had limited it to about one-tenth what it should have been [11]. This mistake, however, only caused network issues during high online traffic on election day.

While technical incidents had transpired in previous Canadian municipal elections, none had been of such magnitude. For one, the 2018 technical incident resulted in greater extensions in voting than had occurred previously. Second, because of the trend to eliminate paper voting, many municipalities did not have a non-electronic option for voters to use, giving those communities no other option but to declare a state of emergency under the *Municipal Elections Act* in order to extend voting eligibility to include the following day.[6] These emergency declarations made national news and sparked discussion about whether online voting uptake in Ontario municipalities would consequently be curbed or halted [12].

2.2 The COVID-19 Pandemic

In Canada, the COVID-19 pandemic was declared a public health emergency in March 2020 when the respiratory illness began to spread, filling hospitals and resulting in a record number of deaths [13]. In response, 80 countries around the world postponed a variety of elections [5], and many more modified delivery or undertook reforms to offer

[6] Declaring a state of emergency is a requirement if a municipality wants to continue an election past election day. 35 municipalities declared a state of emergency in 2018.

regular elections that were accessible to voters. In Canada, the COVID-19 pandemic equally affected how elections were and are run at all levels of government.

At the federal level, Canada's national electoral management body, Elections Canada, approved a series of administrative changes to respond to public health concerns, including implementing physical distancing and other safety guidelines at polling stations, providing all electors with single-use pencils and masks upon entry to a polling station, increasing the capacity of the vote-by-mail system and providing prepaid postage to electors choosing to vote-by-mail, and offering virtual training for election workers [14]. The agency, however, did not consider introducing or mandating either online or telephone voting, citing a significant planning/implementation process constrained by its current operational capacity [14].

For these reasons, calls for early provincial elections in both New Brunswick and British Columbia during the height of the pandemic were met with resistance and public debate about whether elections should be carried out during public emergencies [15]. According to a study conducted by Garnett et al. (2021), between 50% and 60% of respondents agreed that governments, if given the option, should not have called an election during the COVID-19 pandemic.

To mitigate health and safety concerns, provincial election agencies also implemented special voting arrangements, including the adoption of voting-by-mail in New Brunswick [16], British Columbia [17], and Newfoundland and Labrador. Likewise, British Columbia's use of telephone voting saw a significant uptick from previous elections in which such technology was also offered [18]. Given that Saskatchewan's election was held on a fixed date, its EMB prepared several legislative changes to facilitate safe voting, including the modification of mail-in voting requirements and the implementation of additional advance polling opportunities [19]. In addition to these modifications, the COVID-19 pandemic caused provincial election agencies to look more closely at the use of technology in the election process by convening committees [20], developing regulations [21], and/or conducting research [22].

Municipally, elections were also affected by the COVID-19 pandemic. Local elections in New Brunswick, for example, were postponed for over six months in 2020 [23]. While elections held around the same time in Nova Scotia also raised concerns, most of its municipalities used online and telephone voting, which mitigated major delays in the election process. Overall, the COVID-19 pandemic has and continues to serve as a shock to Canadian elections, resulting in modifications or reforms to the voting process.

3 Literature Review

Generally, the machinery of elections tends to be remarkably stable. Policy systems and political processes themselves are often characterized by steadiness and incrementalism, which means drastic change is infrequent. Simply put, reforms to electoral systems, as well as other changes to the structure of elections, are relatively rare occurrences across democracies [24]. Canada is no exception to this trend, despite increasing calls for electoral reforms at national and sub-national levels of government. These calls, which brush up against the longstanding stability of Canadian electoral institutions, have, however, increasingly resulted in smaller scale administrative reform. Reforms to the ways that elections are carried out are often the result of incremental change, although others have also occurred in response to unexpected events. These unexpected events – often referred to as "exogenous shocks" in policy literature – may realign policy systems or policy thinking and are therefore often responsible for the conditions that allow for institutional/organizational change [25–27]. Importantly, many argue that this applies to electoral reform as well [28]. Since the onset of the COVID-19 pandemic, a major exogenous shock, at least 80 countries and territories have postponed national and/or subnational elections [20, 29]. This has led to substantial policy reform, including in Scotland and Wales, where the expeditious passing of the 2020 *Coronavirus Act*[7] legislated the deployment of "emergency powers" to postpone elections to: (1) slow the spread of the virus; (2) reduce resourcing and administrative burden on public bodies; and (3) limit the impact of staffing shortages on the delivery of public services [31].

The *Coronavirus Act* is but one example of the impact that exogenous shocks may have on policymaking processes, which, research suggests, tend to reconfigure policy spaces or subsystems [26]. These circumstances are referred to as "policy windows": moments in time when an issue captures the attention of decision-makers. Policy windows increase the likelihood that different policy and policy initiatives will merge, creating policy action and often breaking a pre-established status, generally under the guise of policy entrepreneurs who recognize opportunity and act accordingly [32, 33].

Much like the COVID-19 pandemic, technical incidents can also act as sizeable shocks to elections. In fact, the more that efforts to modernize elections involve the adoption of technologies [34], the greater the potential for the occurrence of serious technical incidents. The security of electronic voting has raised particular concern about system vulnerabilities [35, 36], authentication and verification issues [37, 38], and electoral fraud [39]. These issues have been identified not only in Canada, but also in Switzerland [40], Estonia [35], Australia [8], Finland [41], and India [42], among others. Some of these technical shocks have resulted in extensive delays and closures of voting booths [43, 44], and, in certain instances, have fueled public skepticism about election integrity [45]. In serious cases, technical shocks have also led jurisdictions to either halt plans for electoral reform (i.e., Switzerland) or to completely abandon intentions to adopt online voting (i.e., United Kingdom, Norway, and Australia). James and Alihodzic (2020) explain that both technical and logistics issues, even when having occurred without the simultaneous presence of other exogenous shocks, have historically resulted in policy change, including during the 2019 Nigerian Presidential Election and the 1996 post-war Bosnian elections.

For these reasons, both the COVID-19 pandemic and the 2018 technical incident can be classified as exogenous shocks that may lead to the opening of policy windows to voting reform. As such, local clerks, politicians, and administrators could be seen as policy actors who have the power to take advantage of policy windows to change course on the implementation of voting methods in certain communities. Given widespread tendencies to postpone or cancel elections in the wake of exogenous shocks and major emergency situations, including natural disasters, war, and military coups [46], policy windows ought to figure centrally in research on electoral administration and reform. In fact, there has been no shortage of articles examining how the COVID-19 pandemic has

[7] The *Coronavirus Act* received royal assent in 2020.

led to fundamental upheaval in several policy domains [47–50]. Under these conditions, policy directions or ideas that once seemed fundamentally unworkable or risky may suddenly become viable.

However, while policy windows create an opportunity for policy action, actors still need to take advantage of those openings. The literature refers to actors who take such advantages as "policy entrepreneurs" [32]. These actors possess the knowledge, power, tenacity, and luck to exploit key opportunities and, by mustering the resources to take advantage of them, can enact crucial policy change [51]. This may explain why changes in policy tend not to be uniform across jurisdictions, even when those jurisdictions experience the same or similar events. Policy entrepreneurship necessarily creates policy or regulatory differentiation, even with the same forces acting upon multiple jurisdictions.

The adoption of online voting across Ontario municipalities is a prime example of regulatory differentiation in action. As a major shift in electoral administration, online voting has gained traction, in certain municipalities, because of its potential to offer benefits to the democratic process, even if it may also pose a significant risk to those same systems.

To date, most of the literature examining online voting focuses on its effects on voters [52], cost-efficiency [53], or security concerns [54, 55]. By comparison, studies focused on online voting deployment in Europe tend to examine voter participation and trends in turnout [56], the impact of such technologies over time [57], and the actors involved in governance and administration [58]. There is limited literature available on the interaction between the adoption of online voting and exogenous shocks, which, as this paper hypothesizes, may exert different pressures on administrators and EMBs to adopt or not adopt electronic voting methods.

4 Data and Methods

This paper takes a mixed methods approach that relies on both quantitative and qualitative analysis. Drawing upon data from a survey and a focus group with municipal elections administrators, we examine whether the COVID-19 pandemic and a 2018 technical incident have served as countervailing forces on local elections in Ontario, Canada to gauge their perceived impact on the administration of those local elections. Within research on municipal elections, insight from administrators is rare. Although they are crucial decision-makers who play a central role in the carrying out of democratic responsibilities, their insight on election processes is understudied. For this reason, this dataset is both novel and important to develop a better understanding of the decision-making processes that inform election administration, and the considerations taken when engaging in small-scale election reforms, including those necessary to respond to exogenous shocks.

The survey informing this study was administered between April 21 and May 27, 2022, to local officials responsible for election governance in Ontario. To identify potential survey respondents, we obtained a contact list from the Association of Municipal Managers, Clerks, and Treasurers of Ontario (AMCTO) which provided the contact information of 682 individuals responsible for the administration of local elections across Ontario's 444 municipalities. A total of 676 valid emails were sent with an invitation to take part in the survey, and two reminder emails were sent thereafter, each one week apart. 281 respondents completed the survey (from 217 municipalities), indicating a response rate of 41%. Surveys were coded and distributed via the Qualtrics interface and included questions regarding the rationale for online voting adoption, including its benefits and challenges, how the 2018 technical incident and the COVID-19 pandemic affected the administration of local elections, and some attitudinal and demographic items. Questions related to the 2018 technical incident and COVID-19 pandemic allowed us to compare the perceived effects (both magnitude and direction) upon the likelihood of adopting or maintaining an online voting option in Ontario's local elections now and into the future.

Our sample includes a good cross-section of municipalities from which administrators run elections: 54.3% of respondents in our sample are from municipalities with populations fewer than 10,000 persons, 34.3% between 10,000 and 99,000 persons, and 11.3% with over 100,000 persons (N = 25). This aligns with the general make-up of Ontario municipalities given that many of them (approximately 70%) have populations fewer than 10,000. This large number of survey respondents from a representative cross-section of Ontario municipalities allows us to approximate the total municipal population more closely. This is a particular strength of this dataset, because it allows us to draw larger conclusions about the effects of the 2018 technical incident and the COVID-19 pandemic on the administration of elections, generally.

Administrators who completed the survey report being experienced in the administration of municipal elections, with only 13% indicating that they had been involved in local elections administration for one year or less. A majority of respondents – 55% – indicated having at least 10 years of experience (the remaining 32% having spent 2 to 9 years in such a role).

To gain additional explanatory insight and augment the open-ended responses posed in the survey, we also carried out a focus group with members from AMCTO's Election Working Group: a consortium of municipal officials responsible for the administration of elections that meet regularly to discuss issues and share best practices and other strategies for election planning, implementation, and evaluation. Prior to taking part, participants were provided with a focus group guide outlining four themes for discussion: (1) the state of electronic voting in Ontario municipalities; (2) voting methods and the 2018 technical incident; (3) voting methods and the COVID-19 pandemic; and (4) the future of local elections in Ontario.

The focus group consisted of six participants from five cities that varied in size and who have different histories of voting method use and opinions on online voting. Two of the five cities had not adopted online voting in their municipality's local election, and likewise indicated having no desire to use the technology in upcoming local elections. One city had not used online voting methods previously but regretted the decision not to include it as a voting option in the 2022 municipal election. The final two communities had previously used online voting and were affected by the 2018 technical incident.

The analysis below proceeds in three parts. Firstly, we consider the perceived impact of both the 2018 technical incident and the COVID-19 pandemic upon decisions to use online voting in the upcoming 2022 municipal elections. We present frequency distributions to survey questions that reveal perceptions of the direction and magnitude of these effects upon the likelihood that municipalities decided to adopt online voting for the upcoming 2022 municipal elections. Second, we merge the direction and magnitude variables (by multiplying them) to create a new, composite indicator that taps into both perceived direction and magnitude of the effects of each of the exogenous shocks. Two composite variables (one for the COVID-19 pandemic and one for the 2018 technical incident) then serve as outcome variables in regression models where they are regressed onto a series of city-level variables: population, urbanity, and online voting usage. Finally, to add explanatory insight to the survey results, we draw upon the focus group discussion and open-ended survey comments.

5 Results

5.1 The 2018 Technical Incident

Survey respondents were asked separate questions about whether the 2018 technical incident and concerns about the COVID-19 pandemic had an effect on the likelihood that online voting was going to be adopted in their municipality in the upcoming 2022 election (response options were ‘more likely’, ‘less likely’, and ‘did not make a difference’). Table 1 contains the frequency distributions for these questions.

Table 1. Perceived direction of impact of shocks upon likelihood of use of online voting

	Technical incident	COVID-19
Less likely	19.2%	0.0%
No effect	80.8%	52.7%
More likely	0.0%	47.3%
N	240	241

Table 1 indicates two notable findings: First, it confirms that the perceived impact of the two shocks are indeed pulling in opposite directions. Roughly one-fifth of administrators thought that the 2018 technical incident decreased the likelihood of adopting online voting, and not a single respondent thought it made it more likely. As for the effects of the COVID-19 pandemic, nearly half of respondents thought it made the use of online voting more likely, and no one replied that it made it ‘less likely’. There are clear directional effects here. Secondly, election administrators were much more likely to believe that the COVID-19 pandemic influenced online voting decisions than the 2018 technical incident. Over 80% reported that the incident had no effect, while fewer than half took this opinion of the COVID-19 pandemic. Combining responses from these questions, we see that respondents were also considerably more likely to say that the COVID-19 pandemic had an effect, but the 2018 technical incident did not, rather than the other way around. 38.5% of respondents thought that the COVID-19 pandemic was the only factor that affected the adoption of online voting, while just 9.7% thought only the 2018 technical incident mattered (8.4% said that both mattered, while 43.4% said that neither did).[8] We therefore see a significant difference in the perceived impact of the two shocks.

One interesting question to consider is whether respondents from municipalities that were affected by the 2018 technical incident perceived the 2018 technical incident to have a greater impact on the likelihood of using online voting than those in municipalities that were not affected. A total of 40 officials from 29 municipalities affected by the 2018 technical incident responded to the survey. The remaining respondents administered elections in municipalities that were not affected by the 2018 technical incident. Interestingly, despite these varied orientations to the 2018 technical incident, we see no statistically significant differences between responses from administrators from the two types of municipalities. Put simply, though the 2018 technical incident only affected a small share of Ontario's municipalities, its perceived effects were just as likely to be felt in unaffected municipalities as they are in those that were directly affected.

Thinking about the magnitude of effects, we can elaborate upon these findings by examining the results of a second set of survey questions that asked respondents *how much* of an impact the two shocks had on the administration of elections (response options were 'a lot', 'somewhat', 'a little', and 'not at all'). Table 2 contains the frequency distributions for these two variables.

Table 2. Perceived magnitude of impact of shocks on decision to use online voting in 2022

	Technical incident	COVID-19
None	71.7%	53.1%
A little	7.1%	9.5%
Somewhat	12.1%	23.7%
A lot	9.2%	13.7%
N	240	241

Table 2 provides further evidence that the COVID-19 pandemic is perceived to have more of an impact than the 2018 technical incident on the municipalities in our sample. Rates of 'a lot' and 'somewhat' responses are considerably higher for the COVID-19 pandemic than the 2018 technical incident. A chi-square test reveals that the differences between these distributions is significant at $p < 0.01$.

Overall, then, we see differences in the perceived effects of these shocks both in terms of direction and magnitude. Survey respondents were considerably more likely to say that the COVID-19 pandemic had an effect upon the decision-making in their municipalities than they were to say the same about the 2018 technical incident. Of these, respondents from communities affected by the 2018 technical incident perceive both shocks to have had a greater impact than those that were unaffected, with the belief that the COVID-19 pandemic had a larger influence on likelihood of use. Respondents were also of the opinion that the pandemic had a stronger impact upon their decisions. Results clearly show that the shocks had different effects.

[8] $N = 180$.

5.2 Do Effects Differ Across Municipal Types?

Having studied impacts of direction and magnitude, we turn to evaluating whether the perceived effects of the two exogenous shocks differed depending upon the context in which decisions on voting methods were made. Municipalities in Ontario have vastly different characteristics which we expect may influence opinions of the perceived impact of the two exogenous shocks. In particular, we expect that three factors might affect these calculations.

First, it is conceivable that the impact of the exogenous shocks might hinge upon whether online voting was in place in a previous election (including and especially during the 2018 municipal election). Municipalities deciding whether to adopt online voting for the first time might be expected to deliberate differently than those who have used such a system in the past. Inertia (or path dependency) is a powerful force in any institution. Second, the population size of a municipality may matter. Larger cities have more resources and may be able to cover the costs of in-person elections more easily. Finally, we consider whether cities are classified as urban, suburban, or rural. One might expect that the density of cities may factor heavily into decisions on voting methods. Though it is conceivable that other contextual factors will matter, given the modest size of our sample, we focus on three contextual factors only. Survey respondents were asked questions that address all these factors, and their responses were used to create a series of dummy variables.

These indicators serve as independent variables in a series of two regression models – one for the 2018 technical incident and another for the COVID-19 pandemic. The dependent variables are the aforementioned composite ‘perceived impact’ variables, calculated by multiplying the ‘direction’ and ‘magnitude’ variables considered directly above. Multiplying these separate variables allows us to create a new, single variable, that taps into both direction and strength of perceived effects. In theory, these variables range from -1, which indicates ‘a lot’ of negative effect (making online voting less likely) to 1 (‘a lot’ of positive effect). A value of 0 indicates no perceived impact. In practice, however, the COVID-19 pandemic variable has only positive values, since no respondents took the view that the COVID-19 pandemic decreased the likelihood of the use of online voting. All values for the 2018 technical incident variable are negative, for the same reason (no respondents thought it made online voting more likely) (See Fig. 1).

Table 3 shows the results of the two OLS regression models. Perhaps the most immediately striking result in the table is that none of the explanatory variables are related to the perceived impact of the COVID-19 pandemic. Neither the system previously used, population size, or urbanity seem to have mattered for deliberations over whether to use online voting in the upcoming 2022 municipal elections. In this instance, the null findings are quite meaningful, as they suggest that the effects of the COVID-19 pandemic on the likelihood of adopting online voting were the same across all types of municipalities. Clearly, the effects of the COVID-19 pandemic are profound and equally wide-reaching.

The null findings are also noteworthy because they are vastly different from those observed for the 2018 technical incident model. Here we see that the perceived effects of the 2018 technical incident are context dependent in two dimensions. First, the perceived impact of the 2018 technical incident upon the likelihood of using online voting is in the positive direction in municipalities that used the system in 2018, as compared to

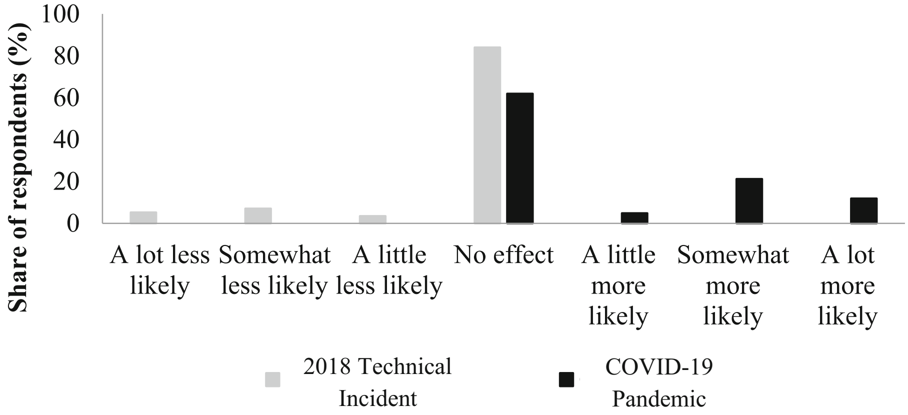


Fig. 1. Frequency distributions for 2018 technical incident and COVID-19 pandemic variables[9]

Table 3. Context and perceived impact of shocks

		Technical incident	COVID-19
	Used online voting in 2018	0.11 (0.04)**	0.02 (0.05)
Baseline = < 10,000	Population >10,000, <100,000	−0.09 (0.04)*	0.09 (0.06)
	Population >100,000	0.07 (0.08)	0.08 (0.11)
Baseline = Rural	Urban	−0.09 (0.06)	−0.08 (0.08)
	Suburban	−0.08 (0.07)	−0.04 (0.09)
	Constant	−0.12 (0.03)**	0.26 (0.04)**
	N	221	223
	Adjusted R2	0.0534	−0.0089

those that did not. Given that the highest value for the outcome variable here is 0, this result can be interpreted to suggest that the incident had a greater perceived effect (in the negative direction) in those cities that had not previously used online voting. This finding aligns with previous research on electoral administration [59] that suggests that administrators tend to favour the electoral systems they have experience with. If taken to be true, this cognitive bias may explain why, in our case, that administrators in cities that have already adopted online voting were less affected by the 2018 technical incident than are those who have no prior experience running an online election.

Population size also appears to have affected deliberations. Medium sized municipalities were affected more (in the negative direction) than either smaller or larger municipalities. Concerns over the 2018 technical incident had a particularly significant

[9] The mean value for the composite 2018 technical incident variable is −0.11, and standard deviation is 0.28. For the COVID-19 pandemic variable, the mean is 0.28, and standard deviation is 0.38.

effect in medium sized municipalities. We suspect that this may stem from medium sized municipalities' greater likelihood of having employed online voting in 2018, meaning that they were more likely to have personally experienced the 2018 technical incident, or to have been particularly sensitive to it. In fact, amongst our sample, rates of use of online voting in 2018 were markedly higher in medium sized municipalities (56.5%, N = 108) than either small (44.4%, N = 69) or large (30.0%, N = 20) municipalities. Given that the small and large groups had relatively lower rates of online voting use, there was less room for the 2018 technical incident to negatively affect the likelihood of using such a system in the upcoming 2022 municipal elections. In contrast, as rates of previous use in medium sized cities were much higher, there was greater potential for the system becoming less likely in 2022.

5.3 Open Ended Responses and Focus Group Insights

To gain additional insight to enhance the survey data results, we consulted responses to two open-ended questions from the survey that asked respondents to provide additional detail about why the 2018 technical incident or the COVID-19 pandemic influenced their decision to either use or not use online voting. To understand the thinking of municipal administrators more deeply, we draw upon feedback provided in the focus group discussion.

The sentiments communicated by the focus group echo and further explain the findings above in three ways. First, participants reported that the 2018 technical incident did not have much of an effect on the administration of their elections. Even administrators from municipalities that were directly affected by the 2018 technical incident commented that they would have no issue hiring the company again and expressed that "there are issues with everything. We had a big issue with mail-in voting kits in 2010."

Another municipal administrator explained that "there's not many options, so you're forced to go with one of the [existing] vendors in the space." Comments suggested that the outage was part and parcel of using technology in elections and not unlike other issues that may arise with paper or mail-in ballots. Some cities that had not used online voting felt similarly. Municipal administrators also emphasized that while the outage affected things, it only impacted the timeliness of the results and not their reliability. These feelings were equally captured in open-ended responses provided in the survey. A comment from one respondent aptly summarizes this perspective: "Over the years [we] have used all methods of voting, and problems have happened with each voting method. Therefore, a problem with electronic voting would not affect my decision on what voting method to use."

On the other hand, for some municipalities that had not used online voting in the 2018 election, the voting outage reinforced – and in some instances strengthened – the justification not to use the voting method, albeit those with that view were in the minority. However, even when the 2018 technical incident was cited, it was not positioned as the main reason for non-adoption. As one administrator whose municipality had never used online voting and has no plans to use it in the future commented, "the 2018 voting incident was a red flag in our report, but it wasn't the main reason [for non-adoption]." Instead, the primary justification to not adopt online voting centered on reliable access to the internet. A quote from one survey respondent clearly captures this point of view:

"Our main reason for no internet voting is lack of Internet infrastructure. The incident would certainly be a learning experience to ensure a better experience in the future." While many of the municipalities with this view are smaller, rural communities, others from large urban centers expressed similar concerns. Overall, the 2018 technical incident was perceived mostly as the "cost of doing business." For a small minority, however, it did result in rolling back adoption, halting further implementation, or abandoning online voting altogether. Four of the communities that took our survey switched back from online to paper ballots.

A second notable finding communicated in the focus group and open-ended survey responses that mirrors the results above is that the COVID-19 pandemic had more of an impact on the adoption of online voting than the 2018 technical incident, although it did not push all communities to adopt the voting mode. The largest group of open-ended comments focused on explaining why municipalities felt compelled to adopt online voting in response to the pandemic. The feelings around doing so were expressed by one respondent who commented that "not knowing at what stage the pandemic would be at election time, we made a point of advising council that internet and telephone voting was immune to the pandemic." Another respondent spoke of some of the challenges other remote voting modes like mail voting can pose: "as an election administrator, this was very important to have included in the 2022 election, even if it wasn't the only method, as there are always issues with Canada Post strikes near elections…"[10] This sentiment was emphasized by the focus group who observed that the COVID-19 pandemic has made it harder to get paper election materials and that the cost of paper ballots has nearly doubled. Clearly, supply-chain issues and reliability of postal service were also considerations that caused some municipalities to lean towards offering online voting.

Many of the administrators who had already used online voting communicated that the COVID-19 pandemic further reinforced their feelings that it was a positive addition to the voting process because of its ability to foster accessibility for electors. Note that many of these respondents represented municipalities that had been affected by the 2018 technical incident. One focus group participant, whose community was likewise impacted by the 2018 technical incident, pointed out that the COVID-19 pandemic had naturally pushed people to use the internet to buy groceries and pay bills, which brought greater public trust and comfort in technology. The administrator expressed that their community was much more "relaxed" about the use of online voting in the upcoming 2022 municipal elections because of the COVID-19 pandemic. However, many municipalities adapted to the pandemic in other ways, and thus did not feel compelled to adopt online voting. These municipalities indicated either introducing or continuing to use mail voting, special ballots, and other precautions, including additional cleaning procedures at polling stations to provide safe and accessible elections.

Finally, the focus group discussions and open-ended responses can help us understand why medium-sized municipalities were more negatively affected by the 2018 technical incident than others. Regarding population size, smaller municipalities either (1) offered online voting but tended to be less concerned about potential technical issues given that their elections are relatively "low stakes", or (2) did not or could not offer online voting because of unstable internet access and other connectivity issues. As one

[10] Canada Post is a Crown corporation that functions as the primary postal operator in Canada.

administrator commented, “there is not sufficient internet coverage in our municipality to make online voting a viable option.” This latter sentiment was a common theme in many comments.

By contrast, larger municipalities either run paper-based votes because: (1) they view their elections as being “high stakes” and therefore have greater concerns about hacking or election interference, or (2) offer online voting, but have more capacity and resources to carry out precautionary technical assessments, audits, and research before implementation. As one administrator noted, “the Dominion incident was disappointing and serious, but avoidable with proper vendor vetting, in our view. Our municipality undertakes a robust and intensive vendor vetting process that leads to confidence in the provider we ultimately choose.” This comment captures the additional knowledge and vetting capacity of larger municipalities with IT teams and staff, compared to smaller municipalities where those services tend to be subcontracted or performed by staff who also hold other roles and responsibilities.

Surprisingly, access to stable internet was also the primary concern of the largest municipality that participated in our focus group. This municipality likewise held strong convictions about online voting security, especially after witnessing other municipalities adopt online voting methods and then switch back to paper ballots. This reversion was articulated as another “red flag” when considering adopting online voting, albeit secondary to worries over internet access. Lastly, for those communities that had not previously used online voting, the 2018 technical incident seemed to reinforce negative impressions of the voting technology. This perspective is perhaps best relayed from the following comment: “online voting was not considered - period.”

6 Discussion and Concluding Thoughts

This paper considers the impact of a 2018 technical incident and the COVID-19 pandemic on the administration of municipal elections in Ontario. In focusing our study on online voting, this paper explores how these shocks exerted pressures on municipal administrators and elections. Using survey and focus group data, we find that the COVID-19 pandemic had a greater perceived effect upon decisions of whether to adopt online voting than the 2018 technical incident. However, municipalities that had not used online voting in 2018 and medium-sized cities were more negatively affected by the 2018 technical incident. Interestingly, our findings also show that the perceived effects of the 2018 technical incident are just as likely to be felt in unaffected municipalities as they are in those that were directly affected. In contrast, the perceived effects of the COVID-19 pandemic did not hinge upon the previous use of online voting, city size, or the urban/rural divide.

It is somewhat surprising that municipalities, particularly those that were directly affected by the 2018 technical incident, did not perceive it to have a greater effect on their likelihood of use and 2022 decision-making. As aforementioned, this could be due to cognitive bias wherein administrators favour the electoral systems that they have experience with. Another explanation for the continued receptiveness to the voting mode, however, could simply be that time eases negative experiences and memories. Some municipal administrators communicated that while there was “no way” local elected officials would have agreed to online voting after the 2018 election, sometime in the four years since then they “seemed to forget about the incident.” Had we surveyed municipal officials immediately following the 2018 election, perceived effects of the 2018 technical incident may have been stronger.

Finally, it is possible that there is, to some extent, a culture of complacency among municipal officials regarding technical issues. Research examining voters’ satisfaction with, and attitudes towards, online voting before and after the 2018 technical incident points to concerns regarding the culture of risk acceptance among Canadian voters [60]. Since administrators often take their cues from voters (focus group members and open-ended survey comments admit to doing so) it is possible that the 2018 technical incident was not perceived as a greater threat because of local bureaucrats’ greater acceptance of technical risk. Such feelings were emphasized in survey comments and the focus group discussion, capturing a perspective that technical issues are bound to happen and that municipalities are accepting of the associated risks. Such patterns have also been observed among the Canadian public, with the public being open to the risk exposure associated with online activities and less reactive to data breaches or security issues that transpire in day-to-day life. While the implementation of technology certainly brings with it the possibility for problems, a key question is whether elections – as a core institution of democracy – should be held to higher standards than other online activities, such as online banking. This culture of complacency could explain why local administrators in Canada seem to have greater risk tolerance than officials in other countries where online voting programs have been halted or canceled.[11]

Given the persistence of the COVID-19 pandemic and the increasing frequency with which technical incidents occur as elections technologies are more widely adopted, our results offer insights for scholars and policymakers, notably that some exogenous shocks may not impact the delivery of elections as much as one might expect, and that despite certain shocks, elections tend to remain relatively stable. Our study also provides avenues for future research, including about the ways that exogenous shocks may impact decisions to adopt or not-adopt voting technologies at other levels of government. Cross-comparative research may also be conducted on the likelihood of online voting adoption in places that have experienced exogenous shocks versus those that have not, and how different shocks have varied implications on voting and other electoral systems. A third area that merits further exploration is the impact of exogenous shocks on public perception of voting technologies and/or willingness to accept policy changes affecting the administration of elections. Policymakers may likewise use this research to better understand the role of exogenous shocks on the policymaking process, and the ways that policy windows create crucial opportunities and support calls for the advancement of electoral reform that may otherwise proceed slowly, if at all.

[11] Interestingly, our data suggest that using online voting even once, in 2018, is sufficient to minimize the negative effect of the 2018 technical incident upon the likelihood of adopting online voting in 2018. We ran an alternate specification of the ‘Technical incident’ model from Table 3, including another dummy variable that indicates experience with online voting previous to 2018 (results not shown but available from the authors). This variable was statistically insignificant. In other words, this ‘culture of complacency’ may require just one election cycle to take hold.

References

1. Alhquist, J., Copelovitch, M., Walter, S.: The political consequences of external economic shocks: evidence from Poland. Am. J. Polit. Sci. **64**(4), 904–920 (2020)
2. Walter, R.: British parliamentary by-elections during the First World War, 1914–18. Parliam. Hist. **37**(3), 415–434 (2018)
3. Stein, R.M.: Election administration during natural disasters and emergencies: Hurricane Sandy and the 2012 election. Elect. Law J. **14**(1), 66–73 (2015)
4. Umberg, T., Rivera Diaz, J.: Election law changes as a result of COVID-19. Calif. J. Polit. Policy **12**(1), 1–3 (2020)
5. International IDEA: Special voting arrangements. International IDEA (2021). https://www.idea.int/data-tools/data/special-voting-arrangements#globalOverview
6. Halderman, J., Teague, V.: The New South Wales iVote system: security failures and verification flaws in a live online election. In: International Conference on E-Voting and Identity, Bern, Switzerland (2015)
7. Gerathy, S.: NSW election confusion, chaos as computer problem shuts several pre-poll centres. ABC News (2019). https://www.abc.net.au/news/2019-03-13/nsw-election-voters-being-turned-away-from-pre-poll-stations/10895954?nw=0
8. Stilgherrian: Flaws found in NSW iVote system yet again. *ZDNet* (2019). https://www.zdnet.com/article/flaws-found-in-nsw-ivote-system-yet-again/
9. Wasim, A.: RTS controversy likely to haunt ECP, Nadra for a long time. *Dawn* (2018). https://www.dawn.com/news/1424394
10. Cardillo, A., Akinyokun, N., Essex, A.: Online voting in Ontario municipal elections: a conflict of legal principles and technology? In: Krimmer, R., et al. (eds.) E-Vote-ID 2019. LNCS, vol. 11759, pp. 67–82. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30625-0_5
11. Dominion Voting Services: Dominion Voting statement regarding internet voting service slowdown affecting Ontario municipalities. Dominion Voting Services (2018)
12. Britneff, B.: Online voting in 51 Ontario municipalities marred by election-day ‘system load issue’. Global News (2018). https://globalnews.ca/news/4585577/ontario-voting-issues/
13. World Health Organization: WHO Coronavirus (COVID-19) dashboard. World Health Organization (2022). https://covid19.who.int
14. Elections Canada: Impact of COVID-19. Elections Canada (2021). https://www.elections.ca/content.aspx?section=med&dir=cor&document=index&lang=e
15. Poitras, J.: Blaine Higgs calls New Brunswick election for Sept. 14, despite pandemic. CBC News (2020). https://www.cbc.ca/news/canada/new-brunswick/possible-election-covid-19-pandemic-1.5689049
16. Elections New Brunswick: Fortieth general provincial election. Elections New Brunswick (2021). https://www.electionsnb.ca/ content/dam/enb/pdf/2020-prov-rpt.pdf
17. Elections British Columbia: COVID-19 and Elections B.C. Elections British Columbia (n.d.). https://elections.bc.ca/news/covid-19-and-elections-bc/
18. Elections British Columbia: Provincial general election: Report of the Chief Electoral Officer. Elections British Columbia (2021). https://www.elections.bc.ca/docs/rpt/2020-provincial-general-election-report.pdf
19. Elections Saskatchewan: Report of the Chief Electoral Officer pursuant to subsection 7(6) of The Election Act, 1996 regarding actions taken during Saskatchewan’s 29th general election. Elections Saskatchewan (2020). https://cdn.elections.sk.ca/upload/ GE29.-SecVII-Rpt-v1.0-FINAL.pdf
20. Elections Ontario: Advisory committee on standards for voting technologies. Elections Ontario. https://www.elections.on.ca/en/about-us/advisory-committee-for-voting-technology-standards.html

21. Elections Northwest Territories: Local authorities elections act and legislation. Municipal and Community Affairs. https://www.maca.gov.nt.ca/en/services/municipal-elections/local-authorities-elections-act-and-legislation
22. Elections British Columbia: COVID-19 and Elections B.C. Elections British Columbia. https://elections.bc.ca/news/covid-19-and-elections-bc/
23. Lyall, L.: Municipal elections at standstill in N.B. amid COVID-19. CTV News (2020). https://atlantic.ctvnews.ca/mobile/municipal-elections-at-standstill-in-n-b-amid-covid-19-1.5112168?cache=szcuxcmsiqb?clipId=89578
24. Norris, P.: Introduction: the politics of electoral reform. Int. Polit. Sci. Rev. **16**(1), 3–8 (1995)
25. Cross, W., Blais, A.: Who selects the party leader? Party Polit. **18**(2), 127–150 (2012)
26. Williams, R.A.: Exogenous shocks in subsystem adjustment and policy change: the credit crunch and Canadian banking regulation. J. Public Policy **29**(1), 29–53 (2009)
27. Harmel, R., Janda, K.: An integrated theory of party goals and party change. J. Theor. Polit. **6**(3), 259–287 (1994)
28. Renwick, A.: The Politics of Electoral Reform: Changing the Rules of Democracy. Cambridge University Press, Cambridge (2010)
29. International IDEA: Global overview of COVID-19: Impact on elections. International IDEA (2022). https://www.idea.int/news-media/multimedia-reports/global-overview-covid-19-impact-elections
30. Wallace, J., Palder, D.: Elections delayed by Coronavirus. Foreign Policy (2020)
31. Institute for Government: Coronavirus Act. Institute for Government (2020). https://www.instituteforgovernment.org.uk/explainers/coronavirus-act
32. Kingdon, J.W.: Agendas, Alternatives and Public Policies. HarperCollins, New York (1995)
33. Herweg, N., Zahariadis, N., Zohlnhöfer, R.: The multiple streams framework: foundations, refinements, and empirical applications. Theor. Policy Process **4**, 17–53 (2018)
34. Goodman, N., Spicer, Z.: Administering elections in a digital age: online voting in Ontario municipalities. Can. Public Adm. **62**(3), 369–392 (2019)
35. Springall, D., et al.: Security analysis of the Estonian internet voting system. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 703–715 (2014)
36. Culnane, C., Eldridge, M., Essex, A., Teague, V.: Trust implications of DDoS protection in online elections. In: Krimmer, R., Volkamer, M., Braun Binder, N., Kersting, N., Pereira, O., Schürmann, C. (eds.) E-Vote-ID 2017. LNCS, vol. 10615, pp. 127–145. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68687-5_8
37. Murray, J., Kiniry, J., Zimmerman, D., Dzieduszycka-Suinat, S.: The future of voting: end-to-end verifiable internet voting specification and feasibility. US Vote Foundation (2015)
38. Schreyen, G., Rich, E.: Security in large-scale internet elections: a retrospective analysis of elections in Estonia, the Netherlands, and Switzerland. IEEE Trans. Inf. Forensics Secur. **4**(4), 729–744 (2009)
39. Teague, V.: Faking an iVote decryption proof. Thinking Cybersecurity (2019). https://www.thinkingcybersecurity.com/iVoteDecryptionProofCheat.pdf
40. Driza Maurer, A.: The Swiss Post/Scytl transparency exercise and its possible impact on internet voting regulation. In: Krimmer, R., et al. (eds.) E-Vote-ID 2019. LNCS, vol. 11759, pp. 83–99. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30625-0_6
41. Heimo, I.O., Fairweather, B.N., Kimppa, K.K.: The Finnish e-voting experiment: what went wrong? ETHICOMP 2010: The "backwards, forwards and sideways" changes of ICT, pp. 290–298 (2010)
42. Wolchok, S., et al.: Security analysis of India's electronic voting machines. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, pp. 1–14 (2010)

43. Garnett, H.A., Bordeleau, J.-N., Harell, A., Stephenson, L.: Canadian provincial elections during the COVID-19 pandemic. International IDEA (2021). https://www.idea.int/sites/default/files/multimedia_reports/canadian-provincial-elections-during-the-covid-19-pandemic-en.pdf
44. Tovey, J.: NSW election: technical problems down online voting and disrupt pre-poll booths. The Guardian (2019). https://www.theguardian.com/australia-news/2019/mar/13/nsw-election-technical-problem-disrupts-early-voting
45. Shah, A.: Pakistan in 2018: theft of an election. Asian Surv. **59**(1), 98–107 (2019)
46. James, T.S., Alihodzic, S.: When is it democratic to postpone an election? Elections during natural disasters, COVID-19, and emergency situations. Elect. Law J. **19**(3), 344–362 (2020)
47. Beland, D., Marier, P.: COVID-19 and long-term care policy for older people in Canada. J. Aging Soc. Policy **32**(4–5), 358–364 (2020)
48. Auener, S., Kroon, D., Wackers, E., van Dulmen, S., Jeurissen, P.: COVID-19: a window of opportunity for positive health care reforms. Int. J. Health Policy Manage. **9**(10), 419–422 (2020)
49. Minkler, M., Griffen, J., Wakimoto, P.: Seizing the moment: policy advocacy to end mass incarceration in the time of COVID-19. Health Educ. Behav. **47**(4), 514–518 (2020)
50. Dupont, C., Oberthur, S., von Homeyer, I.: The COVID-19 crisis: a critical juncture for EU climate policy development? J. Eur. Integr. **42**(8), 1095–1110 (2020)
51. Cairney, P.: Three habits of successful policy entrepreneurs. Policy Polit. **46**(2), 199–215 (2018)
52. Couture, J., Breux, S., Goodman, N.: La vote par internet augmente-t-il la participation électorale? Cyberespace et science politique: De la méthode au terrain, du vir- tuel au reel, pp. 123–148 (2017)
53. Krimmer, R., Duenas-Cid, D., Krivonosova, I., Vinkel, P., Koitmae, A.: How much does an e-vote cost? Compared costs per vote in multichannel elections in Estonia. In: E-Vote-ID 2018 Conference Proceedings, pp. 18–33 (2018)
54. Neumann, P.G.: Security criteria for electronic voting. In: 16th National Computer Security Conference, vol. 29, pp. 478–481 (1993)
55. Essex, A.: Internet voting in Canada: a cyber security perspective. https://www.ourcommons.ca/Content/Committee/421/ERRE/Brief/BR8610535/br-external/EssexAleksander-e.pdf
56. Goodman, N., Stokes, L.: Reducing the cost of voting: an evaluation of internet voting's effect on turnout. Br. J. Polit. Sci. **50**(3), 1155–1167 (2020)
57. Vassil, K., Solvak, M., Vinkel, P., Trechsel, A.H., Alvarez, R.M.: The diffusion of internet voting: usage patterns of internet voting in Estonia between 2005 and 2015. Gov. Inf. Q. **33**(3), 453–459 (2016)
58. Krivonosova, I.: The forgotten election administrator of internet voting: lessons from Estonia. Policy Stud. 1–25 (2021)
59. Moynihan, D.P., Lavertu, S.: Cognitive biases in government: technology preferences in election administration. Public Adm. Rev. **72**(1), 68–77 (2012)
60. Goodman, N., Germann, M., Essex, A., Tieber, A.: Can a technical incident affect voters' attitudes toward e-voting? Evidence from Canada. Working paper (2022)

The Council of Europe's CM/Rec(2017)5 on e-voting and Secret Suffrage: Time for yet Another Update?

Adrià Rodríguez-Pérez[1,2](✉)

[1] Scytl Election Technologies, S.L.U, 08021 Barcelona, Spain
adria.rodriguez@scytl.com
[2] Universitat Rovira i Virgili, 43002 Tarragona, Spain

Abstract. The Council of Europe's Recommendation CM/Rec(2017)5 on e-voting remains the main international legal standard in the field. According to the updated Recommendation, e-voting should respect all the principles for democratic elections. This includes, of course, the principle of secret suffrage. Provisions on secret suffrage are dispersed throughout Rec(2017)5 and its related documents. The main provisions can be found in Section IV of Appendix I, but the principle is also mentioned in several other sections, in the Explanatory Memorandum, and in the Guidelines. A detailed analysis of all these provisions reveals important flaws in the understanding of secret suffrage in (remote) e-voting. Some of the flaws are the result of an inaccurate understanding of secret suffrage, in which this principle is mixed with provisions on personal data protection. In other cases, the flaws are due to analogies being drawn with paper-based voting channels, which prevent the standards from taking stock of the specificities of (remote) e-voting. In this paper I provide a detailed account of these flaws. I also suggest some alternative approaches and wording for the provisions on secret suffrage. Lastly, I discuss the desirability and feasibility of different alternatives regarding the review of Rec(2017)5.

Keywords: Remote electronic voting · International standards · Secret suffrage

1 Introduction

The Council of Europe's Recommendation CM/Rec(2017)5 on e-voting remains the main international legal standard in the field. According to the updated Recommendation, e-voting should respect all the principles for democratic elections (Council of Europe 2017a: para. i). This includes, of course, the principle of secret suffrage: one of the five principles of the European Electoral Heritage, according to the Venice Commission's Code of Good Practice in Electoral Matters (2002). Provisions on secret suffrage are dispersed throughout Re(2017)5 and its related documents: the Explanatory Memorandum and the Guidelines. The main provisions can be found in Section IV of Appendix I, but the principle is also mentioned in several other sections, either directly or indirectly.

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 90–105, 2022.
https://doi.org/10.1007/978-3-031-15911-4_6

A detailed analysis of all these provisions reveals important flaws in the understanding of secret suffrage in (remote) electronic voting.

In this paper, I provide a detailed account of these flaws. I also suggest some alternative approaches and wording for the provisions on secret suffrage in the Recommendation. Lastly, I discuss the desirability and feasibility of different alternatives regarding the review of Rec(2017)5. The focus of the paper is on remote e-voting[1] technologies. These can take many forms and shapes, but they share one characteristic: the devices used to vote (be it a computer or a laptop, a smartphone or even a smart TV) are located remotely from the voting or counting servers, and the connection between the two depends upon the Internet as the voting channel. Because it is remote, internet voting opens the door to voting from uncontrolled environment, raising concerns about the secrecy of the vote.

The next section provides a brief introduction to the Council of Europe's recommendations on e-voting. The goal is to understand the drivers behind the adoption of these standards and their recent update. In Sect. 3, I look more specifically into the provisions on secret suffrage in the updated Recommendation. I look directly at the standards on secret suffrage, but at the same time I also describe direct and indirect references to this principle throughout the Recommendation. Lastly, Sect. 4 addresses the issue at stake: is it necessary to update Rec(2017)5? I suggest two different issues that should be taken into account regarding the current standards. On the one hand, the scope of the provisions on secret suffrage needs to be revisited. The current provisions mix secret suffrage with personal data protection, which is inaccurate. On the other hand, many of the provisions in the Recommendation are still largely based on how secret suffrage is understood in paper-based elections. I argue that in contrast to the aims behind the update, several provisions still fail at specifying how secret suffrage must be protected in (remote) e-voting. Following, the conclusions provide a summary of the main findings and recommendations in the paper.

2 The Council of Europe's Rec(2017)5

To date, the Council of Europe's standards on e-voting remain the main intergovernmental source in the field. While not binding, the Council of Europe's Recommendations have been voluntarily adopted by several member States of the Council of Europe, including Norway (Barrat et al. 2012; Driza Maurer 2014: 112; Stein and Wenda 2014: 106) and Switzerland (Swiss Federal Council 2013: 46). In Estonia, the Supreme Court has also referred to it and in Belgium the Recommendations have been used as a benchmark when evaluating e-voting (Stein and Wenda 2014: 106). For this reason, Robert Stain and Gregor Wenda have argued that the Recommendation "has been the most relevant international document and reference regarding e-voting" (2014: 105). More recently, Ardita Driza Maurer has also acknowledged that "[t]he Council of Europe is the only international organization to have issued recommendations on the regulation of the use of e-voting" (2017: 146). In this section, I look at the origins of the Recommendation, its update, and the main drivers behind this effort. The goal is to understand why and

[1] I use indistinguishably the terms "remote electronic voting", "internet voting", and "online voting" (also in their shorter versions as "remote e-voting" or "i-voting") to refer to e-casting technologies used from remote environments, both controlled and uncontrolled.

how the Recommendation has been updated before looking into its provisions on secret suffrage with more detail.

2.1 The First Council of Europe's Standards on e-voting

The origins of the Recommendation date back to the early 2000. At the initiative of some member states, the Committee of Ministers of the Council of Europe set up a group of experts and adopted, on 30 September 2004, a recommendation on legal, operational and technical requirements for e-voting: Rec(2004)11 (Council of Europe 2004a, b).

Drawing from various regulations governing elections and voting in the Council of Europe's member States, the recommendation only set minimum standards. The 2004 recommendation stressed that "e-voting shall respect all the principles for democratic elections and referendums" (Council of Europe 2004a: i) and "shall be as reliable and secure as democratic elections and referendums which do not involve the use of electronic means" (Council of Europe 2004a: i). Additional guidelines were adopted regarding the certification of remote electronic voting systems (Council of Europe 2010a) and the transparency of e-enabled elections (Council of Europe 2010b), as well as an E-voting handbook on the "key steps in the implementation of e-enabled elections" (Stein and Wenda 2014: 105).

Ten years after its adoption, however, "voices in favour of a formal update [...] gained strength" (Stein and Wenda 2014: 105). For instance, in their evaluation of the Norwegian experience against the 2004 Recommendation, Jordi Barrat i Esteve and Ben Goldsmith concluded that "[t]he recommendations [sic] do not build on existing public international law [...] say little on the legal basis, trying, on the contrary, to cover every possible situation in a technically neutral way. The consequence is sometime vague wording that makes the enforcement of the recommendation more difficult than it should be" (2012: 8). Additionally, Ardita Driza Maurer (2014: 113) also takes note of criticism coming from Douglas Jones (2004), from Margaret McGaley and J. Paul Gibson (2006), and from Andreas Ehringfeld et al. (2010).

2.2 The Road Towards Updated Rec(2017)5

Therefore, "[f]ollowing an informal experts' meeting in Vienna on 19 December 2013, the Committee of Ministers was confronted with the suggestion to formally update the Recommendation in order to keep up with the latest technical, legal and political developments" (Stein and Wenda 2014: 105). It was argued that "[n]ew technological developments and concepts such as in the context of the verifiability of votes, and conclusions from studies and reports, for instance regarding certification, called for addenda or adaptations" (Stein and Wenda 2014: 107).

A study commissioned to Ardita Driza Maurer (2015), and based on a survey among election administrations in the member states of the Council of Europe, identified the following items within the scope of the update: (1) the definition of e-voting, (2) the responsibilities of Electoral Management Bodies, (3) the notion of risk, (4) the structure of the Recommendation, and (5) the categories of requirements. New standards were drafted and approved by an Ad hoc Committee of Experts on Legal, Operational and Technical Standards for e-voting (CAHVE) in November 2016 (Driza Maurer 2017: 147). The

Committee of Ministers of the Council of Europe finally adopted the updated standards as Recommendation CM/Rec(2017)5 of the Committee of Ministers to member States on standards for e-voting on 14 June 2017.

The current definition has been broadened to include e-voting as well as counting machines. Regarding its structure, the current Recommendation consists of three documents: "the Recommendation, which outlines central aspects of e-voting; an Explanatory Memorandum; and guidelines to inform the implementation of provisions in the Recommendation" (Essex and Goodman 2020: 169). Another important innovation is that the Recommendation also introduces the notion of risk. In this sense, "Recommendation ii. Stresses the need to assess risks, namely those specific to e-voting and to adopt appropriate measure to counter them" (Driza Maurer 2017: 154).

Notwithstanding, possibly the most important change in Rec(2017)5 refers to its approach towards e-voting. While the 2004 Recommendation stated that "[e]-voting shall be as reliable and secure as democratic elections and referendums which do not involve the use of electronic means" (Council of Europe 2004a: i), the updated recommendation has dropped this previous comparison (Driza Maurer 2017: 154). The benchmark in Rec(2017)5 "is [the] respect for all principles of democratic elections and referendums" (Driza Maurer 2017: 154). In practice, it means that "standards should be derived directly from the applicable principles" (Driza Maurer 2017: 154).

Since their adoption, the new standards have been welcomed both by members and non-Members states of the Council of Europe. For example, in the explanatory report to the draft law amending the Federal Act on Political Rights, the Swiss Federal Chancellery referenced the updated Recommendation (2018: 22). They argued that the draft legislation was in line with the provisions of the updated Recommendation on verifiability, certification, and risk management. Elsewhere, Aleksander Essex and Nicole Goodman (2020) have been quick to assess to what extent the Council of Europe's approach to regulating e-voting could work in Canada.

3 Secret Suffrage in Rec(2017)5

Therefore, the Council of Europe's Recommendation Rec(2017)5 on e-voting remains the main international legal standard in the field. Having set the stage with the description of its background and update effort, this section will focus more specifically on its provisions on secret suffrage.

The Recommendation offers a definition of secret suffrage in its Explanatory Memorandum. Based on the Venice Commission's Code of Good Practice in Electoral Matters (2002), secret suffrage is summarised as "the voter has the right to vote secretly as an individual, and the state has the duty to protect that right" (Council of Europe 2017b: para. 14). The Recommendation then identifies a set of standards to fulfil this principle. In what follows, I analyse these standards separately. First, I address those standards that are directly related to secret suffrage, which in the Recommendation are included in Section IV of Appendix I. Second, I identify some additional references to secret suffrage throughout the Recommendation and its additional documents.

3.1 Secret Suffrage: Section IV

Section IV in the first Appendix to the Recommendation is entitled secret suffrage and identifies eight standards related to this principle (standards 19 to 26).

The first of these standards provides a general overview about how (remote) e-voting systems must comply with secret suffrage. In this sense, standard No. 19 reads that "[e]-voting shall be organised in such a way as to ensure that the secrecy of the vote is respected at all stages of the voting procedure" (Council of Europe 2017a). This is an umbrella provision on secret suffrage that "sets the general requirement for secrecy of the vote which applies throughout the entire procedure" (Council of Europe 2017b: para. 63). On the one hand, it references "encryption" (Council of Europe 2017b: para. 64), which is a mean to ensure the confidentiality of the vote. On the other, it also notes "that the votes cast are mixed in the electronic ballot box so the order in which they appear at the counting phase does not allow reconstruction of the order in which they arrived" (Council of Europe 2017b: para. 64) as a mechanism to ensure anonymity.

Following, standard No. 20 provides that "[t]he e-voting system shall process and store, as long as necessary, only the personal data needed for the conduct of the e-election" (Council of Europe 2017a). Standards No. 21 and 22 deal with authentication data and voter's registers, respectively, and not with the right to vote secretly. As I will argue below (Sect. 4.1), secret suffrage is different from personal data protection, and therefore these standards should have not been included under Section IV.

Section IV further details four additional standards, on: receipt-freeness (standard No. 23), election fairness (standard No. 24), a provision about the secrecy of previous choices (standard No. 25), and anonymity (standard No. 26). These standards are indeed all related to secret suffrage and touch upon some of the key concerns about secret suffrage in (remote) e-voting.

Standard No. 23 reads that "[a]n e-voting system shall not provide the voter with proof of the content of the vote cast for use by third parties" (Council of Europe 2017a). According to the Explanatory Memorandum, "[t]he aim of this standard is to prevent the breach of vote secrecy as well as vote selling" (Council of Europe 2017b: 70). This standard has been reviewed, corrected, and clarified from the previous Recommendation (Driza Maurer 2017: 155).

According to standard No. 24, "[t]he e-voting system shall not allow the disclosure to anyone of the number of votes cast for any option until after the closure of the electronic ballot box. This information shall not be disclosed to the public after the end of the voting period" (Council of Europe 2017a).

Standard No. 25 reads that "[e]-voting shall ensure that the secrecy of previous choices recorded and erased by the voter before issuing his or her final vote is respected" (Council of Europe 2017a: 6). Therefore, standard No. 25 extends the reach of confidentiality to the "previous choices recorded and erased by the voter before issuing his or her final vote" (Council of Europe 2017a) and granting them "the same protection as the secrecy of the final vote" (Council of Europe 2017b: para. 76). This is important because it highlights certain requirements that may have to be put in place specifically for (remote) e-voting.

Lastly, standard No. 26 reads that "[t]he e-voting process, in particular the counting stage, shall be organised in such a way that it is not possible to reconstruct a link between the unsealed vote and the voter. Votes are, and remain, anonymous" (Council of Europe 2017a: 6).

3.2 Beyond Section IV

Direct References. In addition to the standards which fall all directly under section IV on secret suffrage, the Recommendation also touches upon this principle in regard to standards No. 44, No. 45, and No. 46.

First, standard No. 44 reads that "[i]f stored or communicated outside controlled environments, the votes shall be encrypted" (Council of Europe 2017a). Since this analysis focuses on i-voting (from uncontrolled environments), this standard fully applies. Second, standard No. 45 can be linked to confidentiality and anonymity. This standard sets that "[v]otes and voter information shall be kept sealed until the counting process commences" (Council of Europe 2017a). Therefore, this standard "clarifies the moment where [sic] sealing ends" (Council of Europe 2017b: para. 45).

Lastly, standard No. 46 provides that "[t]he electoral management body shall handle all cryptographic material securely" (Council of Europe 2017a: 8). This provision is key, not only because it is necessary to efficiently guarantee most of the provisions related to secret suffrage, but also because it draws attention to the relevance of operational measures. In this sense, the key-distribution mechanisms described in the Guidelines for the implementation of this standard (Council of Europe 2017c) are of paramount importance to ensure that the confidentiality and anonymity of the votes are preserved. On top of that, this Guideline acknowledges as well that "[t]he private cryptographic keys be [sic] should be generated at a public meeting" (Council of Europe 2017c), bridging the principle of secret suffrage with the requirements for transparency and observation.

Indirect References. Indirect references to secret suffrage can be found in standards No. 6 (related to equal suffrage), in standards No. 16 to No. 18 (in relation to free suffrage), and in standard No. 40 (related to the reliability of the system). While none of those standards deals in principle with secret suffrage, neither directly or indirectly, they reference this principle either in the provisions of the Explanatory Memorandum or in the Guidelines.

Secret and Free Suffrage. Overall, the Explanatory Memorandum and the Guidelines for the standards on free suffrage detail that their provisions should be balanced against the requirements for secret suffrage. More specifically, these standards highlight the need to balance the transparency and auditability of the election with the protection of secret suffrage. First, standard No. 16 reads that "[t]he voter shall receive confirmation by the system that the vote has been cast successfully and that the whole voting procedure has been completed" (Council of Europe 2017a). This provision is completed in the Explanatory Memorandum to the Recommendation, which reads that "[i]t is good practice to accompany these messages with a reminder and instructions to the voter on how to delete traces of the vote if voting was done from an uncontrolled device" (Council of Europe 2017b: para. 58).

Second, standard No. 17 provides that "[t]he e-voting system shall provide sound evidence that each authentic vote is accurately included in the respective election results. The evidence should be verifiable by means that are independent from the e-voting system" (Council of Europe 2017a). For this standard, the Explanatory Memorandum to the Recommendation reads that "it should be possible to audit the evidence to verify its correctness with tools which are external and independent from the e-voting system. To do so, the e-voting system should provide interfaces with comprehensive observation and auditing possibilities, subject to the needs of secrecy and anonymity of the vote" (Council of Europe 2017b: para. 60).

Third, standard No. 18 notes that "[t]he system shall provide sound evidence that only eligible voters' votes have been included in the respective final result. The evidence should be verifiable by means that are independent from the e-voting system" (Council of Europe 2017a). For this standard, the Explanatory Memorandum to the Recommendation adds that "[v]oters and third parties should be able to check that only eligible voters' votes are included in the election result. At the same time counted votes should be anonymous. In the case of internet voting, there exist encryption methods that do not require decoding before votes are counted (homomorphic encryption). Counting can be performed without disclosing the content of encrypted votes" (Council of Europe 2017b: para. 62).

Secret and Equal Suffrage. Provisions on standard No. 6 also call for taking into account the principle of secret suffrage. More specifically, the standard states that "[w]here electronic and non-electronic voting channels are used in the same election or referendum, there shall be a secure and reliable method to aggregate all votes and to calculate the results" (Council of Europe 2017a: 5). In turn, the Explanatory Memorandum to the Recommendation sets that "[w]hen the number of e-votes or of paper votes is particularly small there is the risk that vote secrecy may be violated if the results of those few votes are disclosed. The aggregation method should contain the necessary technical and procedural safeguards to ensure the consolidation of results of the different voting channels before results are disclosed, thus ensuring secrecy. In addition, procedural rules, related namely to personnel intervening in the counting process, should take into account such cases" (Council of Europe 2017b: para. 7).

Secret Suffrage and the Reliability of the System. Lastly, standard No. 40 prescribes that "[t]he electoral management body shall be responsible for the respect for and compliance with all requirements even in the case of failures and attacks. The electoral management body shall be responsible for the availability, reliability, usability and security of the e-voting system" (Council of Europe 2017a). This is an umbrella provision regarding the obligations of election administrations when they introduce (remote) e-voting, which obviously also includes compliance with secret suffrage. The provisions about this standard in the Guidelines will be discussed further in Sect. 4.2. Below.

4 Time for yet Another Update?

Based on the analysis conducted in the previous section, I am of the opinion that not sufficient effort has been put into directly deriving the standards in Appendix I.IV of the Recommendation from the principle of secret suffrage. More specifically, I have identified two fundamental flaws. The first one is linked to the scope of secret suffrage in the Recommendation. In this regard, including data protection provisions under the scope of this principle is totally inadequate because not all personal data processed in an election is related to the secrecy of the vote. Second, and more importantly, the provisions on secret suffrage are still largely based on how this principle is understood in paper-based elections. For this reason, in this section I suggest a new scope and approach to regulate secret suffrage in the Council of Europe's Rec(2017)5.

4.1 The Need for a Clearer Scope

The need for a clearer scope becomes obvious if one takes into account that the provisions in the Recommendation mix the standards of secret suffrage with those of personal data protection. Secondly, some of the Guidelines also seem to point towards an understanding of secret suffrage as being a means to achieve other principles. However, provisions under Appendix I.IV of the Recommendation should all have secret suffrage as an end in itself.

Secret Suffrage and Personal Data Protection. First and foremost, and as I have previously argued (Rodríguez-Pérez 2020: 175), including data protection provisions under the umbrella of secret suffrage is totally inadequate. Votes may be considered personal data in certain circumstances, but personal data is much broader than the legal assets protected by secret suffrage. Therefore, standards No. 20, No. 21, and No. 22 should be moved to another section in the Appendix.

The flawed understanding of the links between secret suffrage and personal data protection can be found in the Explanatory Memorandum. In standard No. 20, the Explanatory Memorandum to the Recommendation specifies that "[d]ata minimisation aims at ensuring data protection and is part of vote secrecy" (Council of Europe 2017b: para. 65). However, secret suffrage and personal data protection are complementary regulations, sometimes overlapping, but under no circumstances one is "part of" the other.

Personal data is much broader than any data that may fall under the scope of secret suffrage. For example, art. 4(1) of the European Union's General Data Protection Regulation (GDPR) defines personal data as "any information relating to an identified or identifiable natural person ('data subject')". Similarly, the Council of Europe's Convention for the Protection of Individuals with regard to Automatic Processing of Personal Data also defines it as "any information relating to an identified or identifiable individual" (art. 2.a).

As a result, personal data protection regulations apply as well to personal data processed about voters, candidates, and even members of the election administration or election observers (Rodríguez-Pérez 2020: 173–175): their names, addresses and contact details, the fact that they belong to a political party or a civil society organisation, etc. are all personal data. In contrast, secret suffrage would deal only with the contents of the vote cast and the conditions in which voters cast them. Thus, data protection is in fact broader than vote secrecy (some aspects of data protection do not deal with the vote at all) and cannot be "part of" it. There is no room for standards No. 20, No. 21 and No. 22 under the provisions on secret suffrage in Appendix I.IV.

On the Publication of Preliminary Results and Secret Suffrage. Standard No. 24 builds on top of standard No. 19 and prescribes the sealing of the votes cast, thus ensuring its confidentiality. Interestingly, the wording of this provision is aimed at preventing the publication of intermediary results, and is not an end in itself. In this regard, the Explanatory Memorandum states that standard No. 24 "aims at preventing the establishing and publication of intermediary results of the e-voting channel" (Council of Europe 2017b: para. 75).

Nevertheless, secret suffrage should be considered and end in itself and not just a means to prevent the publication of intermediate election results. In fact, a ban on the publication of intermediary results seems more geared towards respecting the principle of equal than secret suffrage (since knowing intermediary results would give advantage to later voters over those who have cast their vote earlier). For this reason, and even if the provision in standard No. 24 is accurate, I think that the aim has been misplaced: if its goal is different from ensuring the voter's "right to vote secretly as an individual" (Council of Europe 2017b: para. 14), then it is not aimed at fulfilling the principle of secret suffrage and should be also moved from this section. This would require, in turn, to come up with a new standard on the need to preserve the voters' choices confidentially.

4.2 The Need for a New Approach

Even more concerning that the flawed scope of these provisions is the fact that the Recommendation has also failed at fully mainstreaming its new approach towards e-voting. In this regard, and in spite of the new benchmark being that "e-voting must respect all principles of democratic elections and referendums" (Council of Europe 2017a: para. i), there are many provisions that are still based on analogies to paper-based voting channel.

This constraint becomes self-evident in the (in)direct references to secret suffrage in the Recommendation. For example, the guidelines for the implementation of standard No 40 read that "[f]rom the moment the vote is cast, no one should be able to read or change it or relate the vote to the voter who cast it. This is achieved by the process of sealing the ballot box, and where the ballot box is remote from the voter, by sealing the vote throughout its transmission from voter to ballot box. In some circumstances, sealing has to be done by encryption.

To seal any ballot box, physical and organisational measures are needed. These may include physically locking the box, and ensuring more than one person guards it. In the case of an electronic ballot box, additional measures are necessary, such as access controls, authorisation structures and firewalls.

A vote is sealed when its content has been subject to the measures that ensure that it cannot be read, changed or related to the voter who cast it" [emphasis added] (Council of Europe 2017c).

These provisions basically translate the processes for the counting of the votes cast on paper to (remote) e-voting. First, they claim that votes are anonymous from the moment they are cast, whereas elsewhere the Recommendation itself mentions that anonymity should be guaranteed before the counting stage (see for example standards No. 26 and No. 45). As a matter of fact, this provision mixes anonymity (not being able to relate a vote to the voter who has cast it) with confidentiality (being able to read the vote). Sealing as described in the Guidelines may ensure confidentiality, but not anonymity. Additionally, the Guidelines prescribe specific measures for the "sealing" of the electronic ballot box, which are "additional" to those used for physical ballot boxes. It is unclear whether the same measures can be applied at all, or whether the Recommendation should have prescribed equivalent measures. Lastly, these provisions use vague wordings such as "sealing", which does not mean anything specifically.

Therefore, an alternative approach would be to actually derive the standards from the different dimensions of secret suffrage (Rodríguez-Pérez 2021: 382). Also based on the Venice Commission's Code of Good Practice in Electoral Matters (2002), the Parliamentary Assembly of the Council of Europe (PACE) has identified three main standards in secret suffrage (2007: 5–6):

- Individuality, meaning that each voter makes an individual choice.
- Confidentiality, meaning that only the voter should know how they have voted, and they should be able to make their choices in private.
- Anonymity, meaning that there should not be a link between the vote cast and the identity of the voter who has cast it.

In what follows, I discuss if the current provisions in the Recommendation clearly address these three standards and how they do it.

About Confidentiality in i-voting. Confidentiality is possibly the standards that has been more accurately addressed in Rec(2017)5. In this regard, standard No. 19 enshrines the standard of confidentiality, broadly understood as "the secrecy of the vote" (Council of Europe 2017a). The reference to "encryption" (Council of Europe 2017b: 41) in the Explanatory Memorandum is in this regard paramount, since most of the systems used nowadays ensure the confidentiality of the votes cast with end-to-end encryption. More importantly, standard No. 25 identifies the need to preserve the confidentiality of previous choices, something that is quite unique to (remote) e-voting.

Standard No. 24 could be linked to confidentiality as well, since it calls for preventing the number of votes cast for each option from being known. However, I have already mentioned that the goal of this standard should be confidentiality *as such*, and not to prevent "the establishing and publication of intermediary results of the e-voting channel" (Council of Europe 2017b: para. 75).

However, the main concern regarding the standard of confidentiality is that there are no specific provisions for long-term privacy. In fact, standard No. 19 is meant to apply "throughout the entire procedure: in the pre-voting stage (e.g. transmitting of PINs, or electronic tokens to voters), during the completion of the ballot paper, the casting and transmission of the ballot and during counting and any recounting of the votes" (Council of Europe 2017b: para. 63). Only he Guidelines on Standard No. 40 point briefly towards post-election data processing, by specifying that "[a]ny data retained after the election or referendum period should be stored securely" (Council of Europe 2017c: 40m). Therefore, it is not clear what may happen with the votes after an election is over.

In this regard, it should be noted that current encryption schemes will be vulnerable against quantum computing. In 1994, Peter Shor found an algorithm that could be implemented by a quantum computer to break contemporary encryption algorithms (Hoofnagle and Garfinkel 2022: 166–167). Regardless of when quantum computers may be available to break these algorithms, any data that is published today is vulnerable against future quantum attacks. According to Ward Beullens et al., "[w]hat makes matters worse is that any encrypted communication intercepted today can be decrypted by the attacker as soon as he [sic] has access to a large quantum computer, whether in 5, 10 or 20 years from now" (2021: 28). In my opinion, the Council of Europe's Recommendation could provide some guidance on how to deal with this challenge (or at least envisage that the confidentiality of the data should be ensured also after the election).

About Anonymity in i-voting. Anonymity is also dealt with in Rec(2017)5. Standards No. 19 and No. 26 are the main provisions. Standard No. 26 reads that "[t]he e-voting process, in particular the counting stage, shall be organized in such a way that it is not possible to reconstruct a link between the unsealed vote and the voter" (Council of Europe 2017a: 6). Therefore, the Recommendation already acknowledges that some link may be kept, as it is often the case for i-voting (Council of Europe 2017b: para. 79): until the counting stage, the encrypted vote (sealed in postal voting) is kept to together with some voter identifier to ascertain that all votes have been cast by eligible voters and to ensure that only one vote per votes is counted and included in the final tally. The wording of this provision thus acknowledges that, in contrast to what tends to happen with paper-based voting in polling stations, remote voting channels (be them electronic or not) always tend to link the identity of the voter to the sealed vote. In this regard, the stress that the link cannot be established with the "unsealed vote" does show that there have been some advances in breaking with the analogies.

The problem with anonymity is how the counting processes is described throughout the Recommendation. Therefore, and in spite of the above-mentioned provisions focusing on what should not happen to ensure anonymity (i.e., not having a link with the unsealed vote), Rec(2017) resorts to analogies when describing the steps in the counting procedures. The best example are the provisions in the Explanatory Memorandum for standard No. 26: it prescribes that "[t]he separation [of the information linked to the voter and the votes] has to be made electronically at a predefined stage before counting takes place" (Council of Europe 2017b: para. 79). Moreover, the Explanatory Memorandum for Standard No. 45 draws a straight analogy to paper-based voting channels: "(and by analogy with the physical ballot box), before unsealing, votes are mixed" (Council of Europe 2017b: para. 134).

Interestingly, in a previous provision it has been acknowledged that "[i]n the case of internet voting, there exist encryption methods that do not require decoding before votes are counted (homomorphic encryption). Counting can be performed without disclosing the content of encrypted votes" (Council of Europe 2017b: para. 62). However, the remainder of the Rec(2017)5 and the related documents do not seem to take this possibility into account. In this regard, with homomorphic tallying it is not even necessary to separate the data as prescribed in standards No. 26 and No. 45 at all.

Lastly, it should be considered whether the provisions on secret and equal suffrage could be included as a requirement for anonymity. In this regard, provisions for Standard

No. 6 in the Explanatory Memorandum mention that "[w]hen the number of e-votes or of paper votes is particularly small there is the risk that vote secrecy may be violated if the results of those few votes are disclosed" (Council of Europe 2017b: para. 7). This is not unique to (remote) electronic voting, but electronic means can be seen as more easily ensuring that the number of votes in the result is high enough to prevent anyone from inferring what each voter has voted. For example, the system could have checks preventing the contents of a ballot box from being decrypted if the number of votes it contains is lower than a pre-defined threshold, and automatically aggregate them with the cyphertexts of another ballot box to tally the election results at a higher level.

About Individuality in Remote Electronic Voting. Lastly, individuality is slightly touched upon in Standard No. 23. Standard No. 23 reads that "[a]n e-voting system shall not provide the voter with proof of the content of the vote cast for use by third parties" (Council of Europe 2017a). To ensure individuality in (remote) e-voting, the Explanatory Memorandum identifies some measures, such as "criminal law provisions" (Council of Europe 2017b: para. 71) and informing voters "on the necessity to delete traces of the voting transaction from the device used to cast the vote and on how to do so" (Council of Europe 2017b: para. 73).

This little attention paid to individuality in the Recommendation is quite striking. Specially if one takes into account that one of the main concerns about i-voting is the fact that voters may be forced to vote in a certain way under duress if they vote from uncontrolled environments (Watt 2003; Birch and Watt 2004; Vollan 2006; Enguehard 2010; Buchstein 2015; Manin 2015; Teorell et al. 2016). This concerns have been mitigated in some cases by allowing voters to cancel any vote that they may have cast electronically, either by voting again online or in a polling station. In fact, Estonia and Norway are well-known examples of countries offering such possibility. In contrast, Rec(2017)5 only addresses multiple voting in order to acknowledge this practice. For example, the guidelines on Standard No. 9 prescribe that "[i]f a voter is allowed to cast an electronic vote multiple times, appropriate measures should be taken to ensure that only one vote is counted" (Council of Europe 2017c).

Whereas the Recommendation may not be the right instrument to impose an obligation on states to adopt multiple voting in i-voting, it should at least address this issue more carefully. At the end of the day, the definition of secret suffrage in the Recommendation also sets that "the state has the duty to protect that right [to vote secretly]" (Council of Europe 2017b: para. 14). Notwithstanding, how the state can protect this right when voters cast their vote electronically from uncontrolled environments remains unaddressed.

4.3 The Need for an Update?

In principle, one of the advantages of the updated Recommendation is its three-tiered structure. The new structure allows for distinguishing between principles, recommendations, standards, and requirements. Principles come from various international legal instruments and not from the Recommendation as such. Recommendations are contained in the Recommendation (paragraphs i. to vi.).

Standards are included in the Appendix I to the Rec(2017)5 (Driza Maurer 2017: 150) and can be distinguished between "legal standards" and "technical standards" (Driza Maurer 2017: 152). Legal standards "set objectives that e-voting shall fulfil to conform to the principles of democratic elections" (Driza Maurer 2017: 152), while technical standards "refer to a technical norm, usually in the form of a formal document that established uniform engineering or technical criteria, methods, processes and practices" (Driza Maurer 2017: 152). According to Ardita Driza Maurer, "the Guidelines [...] offer instructions on the implementation of the standards" (2017: 152).

Since they come from different legal sources (principles from international conventions and treaties, national constitutions and formal law; standards from international recommendations and soft-law, and from national material law; and requirements from lower-level regulations), there is in principle a "hierarchy between principles (top), standards (middle) and requirements (bottom of the pyramid)" (Driza Maurer 2017: 152–153).

Another advantage of this layered approach is that it allows for taking stock of rapid technological change. For example, the rationale for the Guidelines is that "they are supposed to evolve frequently to take stock of legal and technical developments" (Driza Maurer 2017: 154). Furthermore, the Recommendation also introduces "a review policy for the Recommendation which is based on the previous practice of biannual meetings" (Driza Maurer 2017: 154). According to Ardita Driza Maurer, these meetings could be used to consider the update of the Guidelines (2017: 154).

Taking into account this new structure, is it possible to identify (at least) three potential future scenarios for the provisions on secret suffrage in Rec(2017)5:

1. Rec(2017)5 is updated to address these flaws. A complete review of the Recommendation, the Explanatory Memorandum, and the Guidelines would allow for moving the provisions on data protection outside the scope of secret suffrage, review the aim of some standards, and accurately assess the wording of all the provisions related to this principle. In this scenario, the assessment should not be limited to secret suffrage: it may be necessary to address potential flaws regarding the provisions on universal, equal, and free suffrage, as well as on the regulatory and organisational requirements, on transparency and observation, etc. Whereas this is the ideal scenario, it is unlikely to happen given that the Recommendation was reviewed just five years ago and that prior shortcomings did not trigger an immediate update either.
2. Rec(2017)5 remains as it is, regardless of its flaws. The alternative is the *status quo*: the Recommendation remains as it is, including with these inconsistencies. This seems unfortunately the most likely scenario, given the fact that the Recommendation's review policy seems to have shifted towards other technologies in the electoral cycle (Council of Europe 2022), rather than providing an actual review mechanism for Rec(2017)5. Since the Recommendation is a voluntary soft-law standard, it is likely that states following this guidance manage to overcome any of Rec(2017)5's flaws when translating the standards into their national legislation.
3. Specific guidelines are adopted on the implementation of Rec(2017)5. A third alternative exists that takes advantage of Rec(2017)5's new review policy. In this scenario, the Recommendation remains as it is, but the Guidelines are reviewed. This seems feasible, but the problem is that the main shortcomings that I have identified

can be found in the Recommendation itself and in the Explanatory Memorandum, which would not be changed. To compensate these shortcomings, the development of specific Guidelines on secret suffrage and remote e-voting could be considered. These Guidelines could develop the provisions in the Recommendation and recognize some of its limitations. Since the provisions on data protection would remain under the umbrella on secret suffrage, specific Guidelines on personal data protection and remote e-voting could be developed as well. This would provide a platform to clarify the scope of personal data protection as being broader than secret suffrage and to identifying and develop the main principles for personal data protection in European data protection law for (remote) e-voting.

5 Conclusions

Provisions on secret suffrage are dispersed throughout Rec(2017)5 and its related documents. The main provisions can be found in Section of Appendix I. IV, but the principle is also mentioned directly or indirectly in several other sections. A detailed analysis of these provisions reveals important flaws in the understanding of secret suffrage in (remote) e-voting. Some of the flaws are the result of an inaccurate understanding of secret suffrage, in which this principle is mixed with provisions on personal data protection. In a similar way, some of the provisions also point towards secret suffrage being a means to achieve other principles, rather than an end on itself. In other cases, the flaws are due to analogies being drawn with paper-based voting channels, which prevent the standards from taking stock of the specificities of (remote) e-voting.

The paper advances potential future scenarios for Rec(2017)5. Among the three potential scenarios, a full update is the more desirable: it is the only option that would allow for rescoping the provisions on secret suffrage, moving the provisions on personal data protection to another section and addressing some of the definitions for the standards in Section IV. However, this alternative is very unlikely. Therefore, and since the current situation could be improved, a better alternative would be to adopt new Guidelines for Rec(2017)5. One set of guidelines would develop the provisions on secret suffrage, identify the three standards in this principle (individuality, confidentiality, and anonymity), and recognize some of the current shortcomings in the Recommendation and the Explanatory Memorandum. A second set could be adopted on personal data protection: to clarify the scope of personal data protection as being broader than secret suffrage and to identifying and develop the main principles for personal data protection in European data protection law for (remote) e-voting.

References

Barrat i Esteve, J., Goldsmith, B.: Compliance with International Standards. Norwegian E-Vote Project. Washington, United States: International Foundation for Electoral Systems (2012)

Beullens, W., et al.: Post-Quantum Cryptography: Current state and quantum mitigation. European Union Agency for Cybersecurity (ENISA) (2021)

Birch, S., Watt, B.: Remote electronic voting: free, fair and secret? Polit. Q. Publishing **75**(1), 60–72 (2004)

Buchstein, H.: Public voting and political modernization. Different views from the nineteenth century and new ideas to modernize voting procedures. In: Elster, J. (ed.) Secrecy and Publicity in Votes and Debates, pp. 15–51. Cambridge University Press, New York (2015)

Council of Europe: Recommendation Rec(2004a)11 of the Committee of Ministers to member states on legal, operational and technical standards for e-voting (2004a)

Council of Europe: Explanatory memorandum to the Recommendation Rec(2004b)11 of the Committee of Ministers to member states on legal, operational and technical standards for e-voting (2004b)

Council of Europe: Guidelines on transparency of e-enabled elections (2010a)

Council of Europe: Certification of e-voting systems: Guidelines for developing processes that confirm compliance with prescribed requirements and standards (2010b)

Council of Europe: Recommendation CM/Rec(2017a)5 of the Committee of Ministers to member States on standards for e-voting (2017a)

Council of Europe: Explanatory Memorandum to Recommendation CM/Rec(2017b)5 of the Committee of Ministers to member States on standards for e-voting (2017b)

Council of Europe: Guidelines on the implementation of the provisions of Recommendation CM/Rec(2017c)5 on standards for e-voting (2017c)

Council of Europe: Committee of Ministers' Guidelines on the use of information and communication technology (ICT) in electoral processes in Council of Europe member States (2022)

Driza Maurer, A.: Ten years Council of Europe Rec(2004)11. Lessons learned and outlook. In: Krimmer, R., Volkamer, M. (eds.) Proceedings of Electronic Voting 2014 (EVOTE2014), pp. 111–117. TUT Press, Tallinn (2014)

Driza Maurer, A.: Report on the Scope and Format of the Update of Rec(2004)11, Council of Europe (2015)

Driza Maurer, A.: Updated European standards for e-voting. In: Krimmer, R., et al. (eds.) Electronic Voting. E-Vote-ID 2017. Lecture Notes in Computer Science, vol. 10615, pp. 127–145. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68687-5_9

Ehringfeld, A., Naber, L., Grechenig, T., Krimmer, R., Traxl, M., Fischer, G.: Analysis of Recommendation Rec(2004)11 based on the experiences of specific attacks against the first legally binding implementation of e-voting in Austria. In: Krimmer, R., Grimm, R. (eds.) Electronic Voting 2010 (EVOTE10). Lecture Notes in Informatics (LNI) - Proceedings Series of the Gesellschaft für Informatik (GI), vol. p-167, pp. 225–237 (2010)

Enguehard, C.: Introduction à l'analyse de chimères technologiques, le cas du vote électronique. Cahiers Droit Sci. Technol. **3**, 261–280 (2010)

Essex, A., Goodman, N.: Protecting electoral integrity in the digital age: developing E-voting regulations in Canada. Election Law J. **19**(2), 162–179 (2020)

European Commission for Democracy Through Law (Venice Commission): Code of Good Practice in Electoral Matters: Guidelines and Explanatory Report. Adopted by the Venice Commission at its 51st and 52nd sessions (Venice, 5–6 July and 18–19 October 2002)

Hoofnagle, C.J., Garfinkel, S.: Law and Policy for the Quantum Age. Cambridge University Press, Cambridge (2022)

Jones, D.: The European 2004 draft e-voting standard: some critical comments (2004). http://homepage.cs.uiowa.edu/~jones/voting/coe2004.shtml

Manin, B.: Why open voting in general elections is undesirable. In: Elster, J. (ed.) Secrecy and Publicity in Votes and Debates, pp. 209–214. Cambridge University Press, New York (2015)

McGaley, M., Gibson, J.P.: A critical analysis of the Council of Europe recommendations on e-voting (2006). https://www.usenix.org/legacy/event/evt06/tech/full_papers/mcgaley/mcgaley.pdf

Parliamentary Assembly of the Council of Europe: Secret ballot – European code of conduct on secret balloting, including guidelines for politicians, observers and voters. Resolution 1590 and Report (2007)

Rodríguez-Pérez, A.: My vote, My (Personal) data: remote electronic voting and the general data protection regulation. In: Krimmer, R., Volkamer, M., Beckert, B., Küsters, R., Kulyk, O., Duenas-Cid, D., Solvak, M. (eds.) E-Vote-ID 2020. LNCS, vol. 12455, pp. 167–182. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-60347-2_11

Rodríguez-Pérez, A.:Regulating internet voting by analogy: does it work? Challenges and concerns for secret suffrage. Krimmer, R. et al. (eds.) Sixth International Joint Conference on Electronic Voting E-Vote-ID 2021, 5–8 October 2021, University of Tartu, pp. 381–382 (2021)

Stein, R., Wenda, G.: The Council of Europe and e-voting: history and impact of Rec(2004)11. In: Krimmer, R., Volkamer, M. (eds.) Proceedings of Electronic Voting 2014 (EVOTE2014), pp. 105–110. Tallinn, TUT Press (2014)

Swiss Federal Chancellery: Modification de la loi fédérale sur les droits politiques (Passage de la phase d'essai à la mise en exploitation du vote électronique). Rapport explicatif pour la procédure de consultation (2018)

Swiss Federal Council: Évaluation de la mise en place du vote électronique (2006–2012) et bases de développement (2013)

Teorell, J., Ziblatt, D., Lehoucq, F.: An introduction to special issue: the causes and consequences of secret ballot reform. Comp. Polit. Stud. **50**(5), 531–554 (2016)

Vollan, K.: Voting in uncontrolled environment and the secrecy of the vote. In: Krimmer, R. (ed.) Electronic Voting 2006 – 2nd International Workshop, Council of Europe, ESF TED, IFIP WG 8.6 and E-Voting, CCpp. 155–169 (2006)

Watt, B.: Human rights and remote voting by electronic means. Representation **39**(3), 197–208 (2003)

Sweeter than SUITE: Supermartingale Stratified Union-Intersection Tests of Elections

Jacob V. Spertus and Philip B. Stark(✉)

Department of Statistics, University of California, Berkeley, CA, USA
{jakespertus,pbstark}@berkeley.edu

Abstract. Stratified sampling can be useful in risk-limiting audits (RLAs), for instance, to accommodate heterogeneous voting equipment or laws that mandate jurisdictions draw their audit samples independently. We combine the union-intersection tests in SUITE, the reduction of RLAs to testing whether the means of a collection of lists are all $\leq 1/2$ of SHANGRLA, and the *nonnegative supermartingale* (NNSM) tests in ALPHA to improve the efficiency and flexibility of stratified RLAs. A simple, non-adaptive strategy for combining stratum-wise NNSMs decreases the measured risk in the 2018 pilot hybrid audit in Kalamazoo, Michigan, USA by more than an order of magnitude, from 0.037 for SUITE to 0.003 for our method. We give a simple, computationally inexpensive, adaptive rule for deciding which stratum to sample next that reduces audit workload by as much as 74% in examples. We also present NNSM-based tests that are computationally tractable even when there are many strata, illustrated with a simulated audit stratified across California's 58 counties.

Keywords: Risk-limiting audit · Election integrity · Supermartingale test · Intersection hypothesis · Multi-armed bandit

1 Introduction

Most U.S. jurisdictions use computers to tabulate votes. Like all computers, vote tabulators are vulnerable to bugs, human error, and deliberate malfeasance—a fact that has been exploited (rhetorically, if not in reality) to undermine trust in U.S. elections [3,4,9,10].

To deserve public trust, elections must be trustworthy, despite relying on untrustworthy software, hardware, and people: they should provide convincing affirmative evidence that the reported winners really won [1,2,20]. Risk-limiting audits (RLAs) are a useful tool for conducting such *evidence-based elections*. RLAs have a specified maximum chance—the *risk limit* α—of not correcting the reported outcome if it is wrong, and never change the reported outcome if it is correct. Below we present methods to reduce the number of ballots that must

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 106–121, 2022.
https://doi.org/10.1007/978-3-031-15911-4_7

be manually inspected in an RLA when the reported outcomes are correct, for stratified audit samples.

In a ballot-level *comparison* RLA, manual interpretations of the votes on randomly sampled ballot cards are compared to their corresponding *cast vote records* (CVRs), the system's interpretation of the votes on those cards. In a *ballot-polling* RLA, votes are read manually from randomly selected cards, but those votes are not compared to the system's interpretation of the cards. All else equal, ballot-level comparison RLAs are more efficient than ballot-polling RLAs, but they require the voting system to export CVRs in a way that the corresponding card can be uniquely identified. Not all voting systems can.

Stratified random sampling can be mandatory or expedient in RLAs. Some states' laws require audit samples to be drawn independently across jurisdictions (e.g., California Election Code § 336.5 and § 15360), in which case the audit sample for any contest that crosses jurisdictional boundaries is stratified. Stratifying on the technology used to tabulate votes can increase efficiency by allowing *hybrid audits* [7,11], which use ballot-level comparison in strata where the voting technology supports it and ballot-polling elsewhere. Another reason to use stratification is to allow RLAs to start before all ballots have been tabulated [17].

The next section briefly reviews prior work on stratified audits. Section 3 introduces notation and stratified risk measurement, then presents our improvements: (i) sharper P-values from new risk-measuring functions; (ii) sequential stratified sampling that adapts to the observed data in each stratum to increase efficiency; and (iii) a computationally efficient method for an arbitrary number of strata. Section 4 evaluates the innovations using case studies and simulations. Section 5 discusses the results and gives recommendations for practice.

2 Past Work

The first RLAs involved stratified batch comparison, using the maximum error across strata and contests as the test statistic [5,13–15], a rigorous but inefficient approach. Higgins et al. [6] computed sharper P-values for the same test statistic using dynamic programming. SUITE [7,11] uses *union-intersection tests* to represent the null hypothesis that one or more reported winners actually lost as a union of intersections of hypotheses about individual strata; it involves optimization problems that are hard to solve when there are more than two strata.

More recently, SHANGRLA [18] has reduced RLAs to a canonical form: testing whether the means of finite, bounded lists of numbers (representing ballot cards) are all less than 1/2, which allows advances in statistical inference about bounded populations to be applied directly to RLAs. Stark [18] showed that union-intersection tests can be used with SHANGRLA to allow *any* risk-measuring function to be used in any stratum in stratified audits.

Stark [19] provided a new approach to union-intersection tests using nonnegative supermartingales (NNSMs): *intersection supermartingales*, which open the possibility of reducing sample sizes by adaptive *stratum selection* (using the first

t sampled cards to select the stratum from which to draw the $(t+1)$th card). Stark [19] does not provide an algorithm for stratum selection or evaluate the performance of the approach; this paper does both.

3 Stratified Audits

We shall formalize stratified audits using the SHANGRLA framework [18], which unifies comparison and polling audits. We then show how to construct a stratified comparison audit using SHANGRLA, how to measure the risk based on a stratified sample, and how adaptive sequential stratified sampling can improve efficiency.

3.1 Assorters and Assertions

Ballot cards are denoted $\{b_i\}_{i=1}^N$. An assorter A assigns a number $A(b_i) \equiv x_i \in [0, u]$ to ballot card b_i [18] and the value $A(c_i)$ to CVR i. The value an assorter assigns to a card depends on the votes on the card, the social choice function, and possibly on the machine interpretation of that card and others (for comparison audits). Stark [18] describes how to define a set of assorters for many social choice functions (including majority, multiwinner majority, supermajority, Borda count, approval voting, all scoring rules, D'Hondt, STAR-Voting, and IRV) such that the reported winner(s) really won if the mean of every assorter in the set is greater than 1/2. The claim that an assorter mean is $> 1/2$ is called an *assertion*. An RLA with risk limit α confirms the outcome of a contest if it rejects the *complementary null* that the assorter mean is $\leq 1/2$ at significance level α for every assorter relevant to that contest.

In a stratified audit, the population of ballot cards is partitioned into K disjoint *strata*. Stratum k contains N_k ballot cards, so $N = \sum_k N_k$. The *weight* of stratum k is $w_k := N_k/N$; the weight vector is $\boldsymbol{w} := [w_1, ..., w_K]^T$. For each assorter A there is a set of assorter values $\{x_i\}_{i=1}^N$. Each assorter may have its own upper bound u_k in stratum k.[1] The true mean of the assorter values in stratum k is μ_k; $\boldsymbol{\mu} := [\mu_1, ..., \mu_K]^T$. The overall assorter mean is

$$\mu := \frac{1}{N}\sum_{i=1}^{N} x_i = \sum_{k=1}^{K} \frac{N_k}{N}\mu_k = \boldsymbol{w}^T\boldsymbol{\mu}.$$

Let $\boldsymbol{\theta} = [\theta_1, ..., \theta_K]^T$ with $0 \leq \theta_k \leq u_k$. A single *intersection null* is of the form $\boldsymbol{\mu} \leq \boldsymbol{\theta}$, i.e., $\cap_{k=1}^K \{\mu_k \leq \theta_k\}$. The *union-intersection form* of the *complementary null* that the outcome is incorrect is:

$$H_0 : \bigcup_{\boldsymbol{\theta}:\boldsymbol{w}^T\boldsymbol{\theta} \leq \frac{1}{2}} \bigcap_{k=1}^{K} \{\mu_k \leq \theta_k\}. \tag{1}$$

[1] The notation we use does not allow u to vary by draw, but the theory in Stark [19] permits it, and it is useful for batch-comparison audits.

From stratum k we have n_k samples $X_k^{n_k} := \{X_{1k}, ..., X_{n_k k}\}$ drawn by simple random sampling, with or without replacement, independently across strata. Section 3.3 shows how to use single-stratum hypothesis tests (of the the null $\mu_k \leq \theta_k$) to test (1). First, we show how to write stratified comparison audits in this form.

3.2 Stratified Comparison Audits

In SHANGRLA, comparison audits involve translating the original assertions about the true votes into assertions about the reported results and discrepancies between the true votes and the machine's record of the votes [18, Section 3.2]. For each assertion, the corresponding *overstatement assorter* assigns ballot card b_i a bounded, nonnegative number that depends on the votes on that card, that card's CVR, and the reported results. The original assertion is true if the average of the overstatement assorter values is greater than 1/2.

We now show that for stratified audits, the math is simpler if, as before, we assign a nonnegative number to each card that depends on the votes and reported votes, but instead of comparing the average of the resulting list to 1/2, we compare it to a threshold that depends on the hypothesized stratum mean θ_k.

Let u_k^A be the upper bound on the original assorter for stratum k and $\omega_{ik} := A(c_{ik}) - A(b_{ik}) \in [-u_k^A, u_k^A]$ be the *overstatement* for the ith card in stratum k, where $A(c_{ik})$ is the value of the assorter applied to the CVR and $A(b_{ik})$ is the value of the assorter for the true votes on that card. Let $\bar{A}_k^b$, $\bar{A}_k^c$, and $\bar{w}_k = \bar{A}_k^c - \bar{A}_k^b$ be the true assorter mean, reported assorter mean, and average overstatement, all for stratum k.

For a particular $\boldsymbol{\theta}$, the intersection null claims that in stratum k, $\bar{A}_k^b \leq \theta_k$. Adding $u_k^A - \bar{A}_k^c$ to both sides of the inequality yields

$$u_k^A - \bar{\omega}_k \leq \theta_k + u_k^A - \bar{A}_k^c.$$

Letting $u_k := 2u_k^A$, take $B_{ik} := u_k^A - \omega_{ik} \in [0, u_k]$ and $\bar{B}_k := \frac{1}{N_k} \sum_{i=1}^{N_k} B_{ik}$. Then $\{B_{ik}\}$ is a bounded list of nonnegative numbers, and the assertion in stratum k is true if $\bar{B}_k > \beta_k := \theta_k + u_k^A - \bar{A}_k^c$, where all terms on the right are known. Testing whether $\bar{B} \leq \beta_k$ is the canonical problem solved by ALPHA [19]. The intersection null can be written

$$\bar{B}_k \leq \beta_k \quad \text{for all} \quad k \in \{1, \ldots, K\}.$$

Define $\boldsymbol{u} := [u_1, \ldots, u_K]^T$. As before, we can reject the complementary null if we can reject *all* intersection nulls $\boldsymbol{\theta}$ for which $\mathbf{0} \leq \boldsymbol{\theta} \leq \boldsymbol{u}$ and $\boldsymbol{w}^T \boldsymbol{\theta} \leq 1/2$.

3.3 Union-intersection Tests

A union-intersection test for (1) combines evidence across strata to see whether any intersection null in the union is plausible given the data, that is, to check whether the P-value of any intersection null in the union is greater than the risk limit.

Consider a fixed vector $\boldsymbol{\theta}$ of within-stratum nulls. Let $P(\boldsymbol{\theta})$ be a valid P-value for the intersection null $\boldsymbol{\mu} \leq \boldsymbol{\theta}$. Many functions can be used to construct $P(\boldsymbol{\theta})$ from tests in individual strata; two are presented below. We can reject the union-intersection null (1) if we can reject the intersection null for all feasible $\boldsymbol{\theta}$ in the half-space $\boldsymbol{w}^T\boldsymbol{\theta} \leq 1/2$. Equivalently, $P(\boldsymbol{\theta})$ maximized over feasible $\boldsymbol{\theta}$ is a P-value for (1):

$$P^* := \max_{\boldsymbol{\theta}} \; \{P(\boldsymbol{\theta}) : \mathbf{0} \leq \boldsymbol{\theta} \leq \boldsymbol{u} \;\text{ and }\; \boldsymbol{w}^T\boldsymbol{\theta} \leq 1/2\}.$$

This method is fully general in that it can construct a valid P-value for (1) from stratified samples and any mix of risk-measuring functions that are individually valid under simple random sampling. However, the tractability of the optimization problem depends on the within-stratum risk-measuring functions and the form of P used to pool risk. So does the efficiency of the audit.

We next give two valid combining rules $P(\boldsymbol{\theta})$. Section 3.6 presents some choices for within-stratum risk measurement to construct $P(\boldsymbol{\theta})$.

3.4 Combining Functions

Ottoboni et al. [11] and Stark [18] calculate P for the intersection null using Fisher's combining function. Let $p_k(\theta_k)$ be a P-value for the single-stratum null $H_{0k} : \mu_k \leq \theta_k$. Define the pooling function

$$P_F(\boldsymbol{\theta}) := 1 - \chi^2_{2K}\left(-2\sum_{k=1}^{K} \log p_k(\theta_k)\right),$$

where χ^2_{2K} is the CDF of the chi-squared distribution with 2K degrees of freedom. The term inside the CDF, $-2\sum_{k=1}^{K} \log p_k(\theta_k)$, is Fisher's combining function[2]. Because samples are independent across strata, $\{p_k(\theta_k)\}_{k=1}^{K}$ are independent random variables, so Fisher's combining function is dominated by the chi-squared distribution with $2K$ degrees of freedom [11]. The maximum over $\boldsymbol{\theta}$, P_F^*, is a valid P-value for (1).

3.5 Intersection Supermartingales

Stark [19] derives a simple form for the P-value for an intersection null when supermartingales are used as test statistics within strata. Let $M^k_{n_k}(\theta_k)$ be a supermartingale constructed from n_k samples drawn from stratum k when the null $\mu_k \leq \theta_k$ is true. Then the product of these supermartingales is also a

[2] Other combining functions could be used, including Liptak's or Tippett's. See Chap. 4 of Pesarin and Salmaso [12].

supermartingale under the intersection null, so its reciprocal (truncated above at 1) is a valid P-value [19,23]:

$$P_M(\boldsymbol{\theta}) := 1 \wedge \prod_{k=1}^{K} M_{n_k}^k(\theta_k)^{-1}.$$

Maximizing $P_M(\boldsymbol{\theta})$ (equivalently, minimizing the intersection supermartingale) yields P_M^*, a valid P-value for (1).

3.6 Within-Stratum *P*-values

The class of within-stratum P-values that can be used to construct P_F is very large, but P_M is limited to functions that are supermartingales under the null. Possibilities include:

- **SUITE**, which computes P_F^* for two-stratum hybrid audits. The P-value in the CVR stratum uses the MACRO test statistic [16]; the P-value in the no-CVR stratum takes a maximum over many values of Wald's SPRT indexed by a nuisance parameter representing the number of non-votes in the stratum. The maximations in MACRO and over a nuisance parameter in the SPRT make SUITE less efficient than newer methods based on SHANGRLA [18].
- **ALPHA**, which constructs a betting supermartingale as in Waudby-Smith and Ramdas [22], but with an alternate parameterization [19]. Such methods are among the most efficient for RLAs [19,23], but the efficiency depends on how the tuning parameter τ_{ik} is chosen. Stark [19] offers a sensible strategy based on setting τ_{ik} to a stabilized estimate of the true mean μ_k. We implement that approach and a modification that is more efficient for comparison audits. Both P_M^* and P_F^* can be computed from stratum-wise ALPHA supermartingales. However, finding the maximum P-value over the union is prohibitively slow when $K > 2$.
- **Empirical Bernstein** (EB), which is a supermartingale presented in Howard et al. [8] and Waudby-Smith and Ramdas [22]. Although they are generally not as efficient as ALPHA and other betting supermartingales [22], EB supermartingales have an exponential analytical form that makes $\log P_M(\boldsymbol{\theta})$ or $\log P_F(\boldsymbol{\theta})$ linear or piecewise linear in $\boldsymbol{\theta}$. Hence, P_M^* and P_F^* can be computed quickly for large K by solving a linear program.

We compare the efficiency of these risk-measuring functions in Sects. 4.1 and 4.2.

3.7 Sequential Stratum Selection

The use of sequential sampling in combination with stratification presents a new possibility for reducing workload: sample more from strata that are providing evidence against the intersection null and less from strata that are not helping. To set the stage, suppose we are conducting a ballot-polling audit with two strata of equal size and testing the intersection null $\boldsymbol{\theta} = [0.25, 0.75]^T$. We have drawn 50 ballot cards from each stratum and found sample assorter means of $[0.5, 0.6]^T$. Given the data, it seems plausible that drawing more samples from the first stratum will strengthen the evidence that $\mu_1 > 0.25$, but additional sampling from the second stratum might not provide evidence that $\mu_2 > 0.75$: to reject the intersection null, it might help to draw disproportionately from the first stratum. Perhaps suprisingly, such adaptive sampling yields valid inferences when the P-value is constructed from supermartingales and the stratum selection function depends only on past data. We now sketch why this is true.

For $t \in \mathbb{N}$ and a particular vector of hypothesized stratum means $\boldsymbol{\theta}$, let

$$\kappa_t(\boldsymbol{\theta}) \in \{1, ..., K\}$$

denote the stratum from which the t-th sample was drawn for testing the hypothesis $\boldsymbol{\mu} \leq \boldsymbol{\theta}$. We call $\kappa(\boldsymbol{\theta}) := (\kappa_t(\boldsymbol{\theta}))_{t \in \mathbb{N}}$ the *stratum selector* for null $\boldsymbol{\theta}$. Crucially, $\kappa(\boldsymbol{\theta})$ is a *predictable sequence* with respect to $(X_t)_{t \in \mathbb{N}}$ in the sense that $\kappa_t(\boldsymbol{\theta})$ can depend on $X^{t-1} := \{X_1, \ldots, X_{t-1}\}$ but not on X_i for $i \geq t$; it could be deterministic given X^{t-1} or may also depend on auxiliary randomness.

For example, a stratum selector could ignore past data and select strata in a deterministic round-robin sequence or at random with probability proportional to stratum size. Alternatively, a rule might select strata adaptively, for instance picking a stratum at random with probability proportional to the current value of each within-stratum supermartingale, so that strata with larger $M^k_{t_k}(\theta_k)$ are more likely to be chosen—an "exploration–exploitation" strategy. In what follows we suppress the dependence on $\boldsymbol{\theta}$ except when it is explicitly required for clarity.

Now, let $M_t^\kappa(\boldsymbol{\theta}) := \prod_{i=0}^t Z_i$ be the test statistic for testing the null hypothesis that the vector of stratumwise means is less than or equal to $\boldsymbol{\theta}$. This is a supermartingale if the individual terms Z_i satisfy a simple condition. Let $Z_0 = 1$ and $Z_i \geq 0$ for all i. If

$$\mathbb{E}_{\boldsymbol{\theta}}[Z_t | X^{t-1}] \leq 1, \tag{2}$$

then $(M_t^\kappa(\boldsymbol{\theta}))_{t \in \mathbb{N}_0}$ is a nonnegative supermartingale starting at 1 under the null. By Ville's inequality [21], the thresholded inverse $(1 \wedge M_t^\kappa(\boldsymbol{\theta})^{-1})_{t \in \mathbb{N}_0}$ is an anytime P-value sequence when $\boldsymbol{\mu} \leq \boldsymbol{\theta}$.

Condition (2) holds if the Z_i are terms extracted from a set of within-stratum supermartingales using a predictable stratum selector: Let

$$\nu_t^\kappa := \#\{i \leq t : \kappa_i = \kappa_t\} \tag{3}$$

be the number of draws from stratum k as of time t. Suppose that for $k \in \{1, \ldots, K\}$, $M_t^k(\theta_k) := \prod_{i=1}^t Y_i^k(\theta_k)$ is a nonnegative supermartingale starting at 1 when X_{ik} is the ith draw from stratum k and the kth stratum mean is $\mu_k \leq \theta_k$. Then if

$$Z_i := Y^{\kappa_i}_{\nu_i^\kappa}(\theta_{\kappa_i}), \tag{4}$$

condition (2) holds and the interleaved test statistic $M_t^\kappa(\boldsymbol{\theta})$ is an intersection supermartingale under the null. We compare two stratum selection rules in Sect. 4.1.

4 Evaluations

4.1 Combination and Allocation Rules

We simulated a variety of two-stratum ballot-level comparison audits at risk limit $\alpha = 5\%$, with assorters defined as in Sect. 3.2. The strata each contained $N_k = 1000$ ballot cards, all with valid votes. Cards were sampled without replacement. The stratum-wise true margins were $[0\%, 20\%]$, $[0\%, 10\%]$ or $[0\%, 2\%]$, corresponding to global margins of 10%, 5%, and 1%, respectively. Stratum-wise reported margins were also $[0\%, 20\%]$, $[0\%, 10\%]$ or $[0\%, 2\%]$, so error was always confined to the second stratum. Each reported margin was audited against each true margin in 300 simulations. Risk was measured by ALPHA or EB combined either as intersection supermartingales (P_M^*) or with Fisher's combining function (P_F^*), with one of two stratum selectors: proportional allocation or lower-sided testing.

In proportional allocation, the number of samples from each stratum is in proportion to the number of cards in the stratum. Allocation by lower-sided testing involves testing the null $\mu_k \geq \theta_k$ sequentially at level 5% using the same supermartingale (ALPHA or EB) used to test the main (upper-sided) hypothesis of interest. This allocation rule ignores samples from a given stratum once the lower-sided hypothesis test rejects, since there is strong evidence that the null is true in that stratum. This "hard stop" algorithm is unlikely to be optimal, but it leads to a computationally efficient implementation and illustrates the potential improvement in workload from adaptive stratum selection.

Tuning parameters were chosen as follows. ALPHA supermartingales were specified either with τ_{ik} as described in Stark [19, Section 2.5.2] (ALPHA-ST, "shrink-truncate") or with a strategy that biases τ_{ik} towards u_k: (ALPHA-UB, "upward bias"). The ALPHA-UB strategy helps in comparison audits because the distribution of assorter values consists of a point mass at $u_A^k = u_k/2$ and typically small masses (with weight equal to the overstatement rates) at 0 and another small value. This concentration of mass makes it advantageous to bet more aggressively that the next draw will be above the null mean; that amounts to biasing τ_{ik} towards the upper bound u_k. Before running EB, the population and null were transformed to [0,1] by dividing by u_k. The EB supermartingale parameters λ_{ik} were then specified following the "predictable mixture" strategy [22, Section 3.2], truncated to be below 0.75. Appendix A gives more details of the ALPHA-ST and ALPHA-UB strategies and the computations.

Sample size distributions for some combinations of reported and true margins are plotted in Fig. 1 as (simulated) probabilities of stopping at or before a given sample size. Table 1 gives estimated expected and 90th percentile sample sizes for each scenario and method. Table 2 lists aggregate scores, computed by finding the ratio of the workload for each method over the smallest workload in each scenario, then averaging over scenarios by taking the geometric mean of these ratios.

Intersection supermartingales tend to dominate Fisher pooling unless the stratum selector is chosen poorly (e.g., the bottom-right panel of Fig. 1 and the

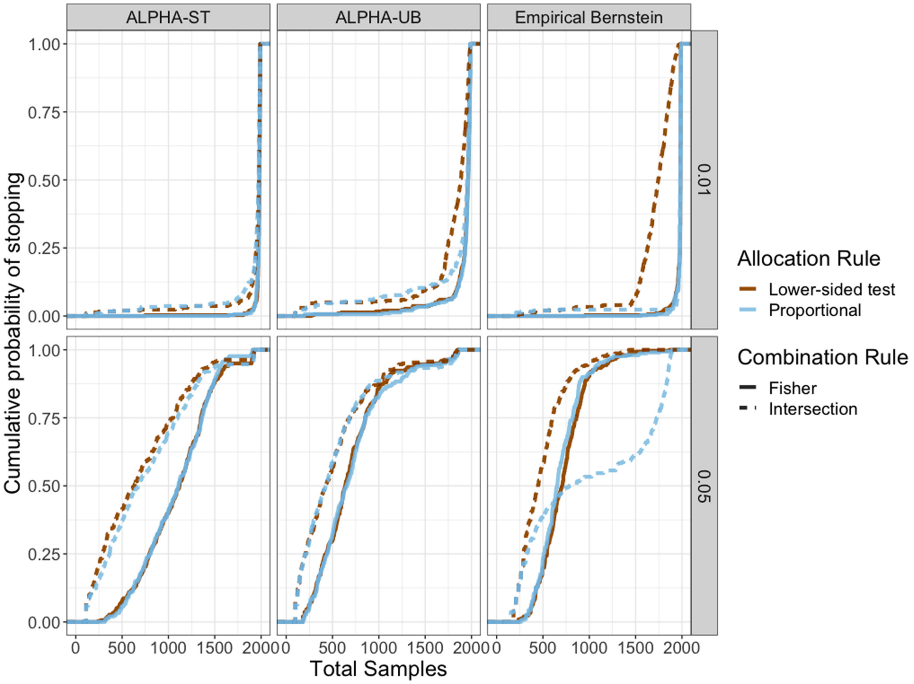


Fig. 1. Probability that the audit will stop (y-axis) at or before different given sample sizes (x-axis) under different allocation rules (indicated by line color: orange for lower-sided testing and blue for proportional allocation) for different combining functions (indicated by line type: solid for Fisher's combining function and dashed for the intersection supermartingale) at risk limit $\alpha = 5\%$. The true margins are in the rows (1% or 5%) while the reported margin is always 10%. Overstatement errors are confined to one stratum. ALPHA-ST = ALPHA with shrink-truncate τ_{ik}; ALPHA-UB = ALPHA with τ_{ik} biased towards u_k.

last row of Table 2). Stratum selection with the lower-sided testing procedure is about as efficient as proportional allocation for the ALPHA supermartingales, but far more efficient than proportional allocation for EB. The biggest impact of the allocation rule occurred for EB combined by intersection supermartingales when the reported margin was 0.01 and the true margin was 0.1: proportional allocation produced an expected workload of 752 cards, while lower-sided testing produced an expected workload of 271 cards—a 74% reduction. Table 2 shows that ALPHA-UB with intersection supermartingale combining and lower-sided testing is the best method overall; ALPHA-UB with intersection combining and proportional allocation is a close second; EB with intersection combining and lower-sided testing is also relatively sharp; ALPHA-ST with Fisher combining is least efficient.

We also ran simulations at risk limits 1% and 10%, which did not change the relative performance of the methods. However, compared to a 5% risk limit,

Table 1. Expected and 90th percentile sample sizes for various risk-measurement functions, reported margins, and true margins, estimated from 300 simulated audits at risk-limit $\alpha = 5\%$. The best result for each combination of reported margin, true margin, and summary statistic is highlighted. Comparison audit sample sizes are deterministic when there is no error, so the expected value and 90th percentile are equal when the reported and true margins are equal.

Reported margin	**supermartingale**	**Combination**	**Allocation rule**	**True margin**					
				0.01		**0.05**		**0.1**	
				Mean	**90th**	**Mean**	**90th**	**Mean**	**90th**
0.01	ALPHA-ST	Fisher	Lower-sided test	1970	1970	1011	1274	338	506
			Proportional	1970	1970	1009	1274	338	540
		Intersection	Lower-sided test	1940	1940	558	848	181	284
			Proportional	1940	1940	554	835	182	298
	ALPHA-UB	Fisher	Lower-sided test	1402	1402	544	754	252	360
			Proportional	1402	1402	548	748	248	354
		Intersection	Lower-sided test	1106	1106	344	**504**	149	238
			Proportional	1106	1106	**342**	510	**148**	**232**
	Empirical Bernstein	Fisher	Lower-sided test	1438	1438	649	768	384	498
			Proportional	1438	1438	647	782	376	464
		Intersection	Lower-sided test	**1102**	**1102**	478	652	271	378
			Proportional	**1102**	**1102**	982	1856	752	1728
0.05	ALPHA-ST	Fisher	Lower-sided test	1973	1986	908	908	305	426
			Proportional	1972	1984	908	908	298	412
		Intersection	Lower-sided test	1930	1980	428	428	145	212
			Proportional	1933	1982	428	428	151	228
	ALPHA-UB	Fisher	Lower-sided test	1769	1970	428	428	217	292
			Proportional	1769	1972	428	428	217	288
		Intersection	Lower-sided test	1611	1884	**256**	**256**	**122**	**176**
			Proportional	1651	1962	**256**	**256**	**122**	180
	Empirical Bernstein	Fisher	Lower-sided test	1882	1986	448	448	306	356
			Proportional	1870	1986	448	448	304	354
		Intersection	Lower-sided test	**1610**	**1858**	296	296	199	234
			Proportional	1924	1982	296	296	302	376
0.10	ALPHA-ST	Fisher	Lower-sided test	1971	1990	1088	1536	240	240
			Proportional	1974	1990	1080	1509	240	240
		Intersection	Lower-sided test	1910	1991	694	1312	112	112
			Proportional	1894	1988	755	1347	112	112
	ALPHA-UB	Fisher	Lower-sided test	1904	1984	696	1107	180	180
			Proportional	1914	1984	715	1263	180	180
		Intersection	Lower-sided test	1756	1968	521	1046	**98**	**98**
			Proportional	1804	1990	534	1079	**98**	**98**
	Empirical Bernstein	Fisher	Lower-sided test	1968	1988	716	987	238	238
			Proportional	1974	1988	686	928	238	238
		Intersection	Lower-sided test	**1697**	**1901**	**487**	**799**	154	154
			Proportional	1939	1990	1000	1846	154	154

a 10% risk limit requires counting about 17% fewer cards and a 1% risk limit requires about 38% more, on average across scenarios and methods.

Table 2. Score for each method: the geometric mean of the expected workload over the minimum expected workload in each scenario. A lower score is better: a 1.00 would mean that the method always had the minimum expected workload. The best score is highlighted. A score of 2 means that workloads were twice as large as the best method, on average, across simulations and scenarios.

supermartingale	**Combination**	**Allocation**	**Score**
ALPHA-ST	Fisher	Lower-sided test	2.11
		Proportional	2.10
	Intersection	Lower-sided test	1.35
		Proportional	1.37
ALPHA-UB	Fisher	Lower-sided test	1.47
		Proportional	1.48
	Intersection	Lower-sided test	1.01
		Proportional	1.02
Empirical Bernstein	Fisher	Lower-sided test	1.73
		Proportional	1.71
	Intersection	Lower-sided test	1.25
		Proportional	1.78

4.2 Comparison to SUITE

SUITE was used in a pilot RLA of the 2018 gubernatorial election in Michigan [7]. Three jurisdictions—Kalamazoo, Rochester Hills, and Lansing—were audited, but only Kalamazoo successfully ran a hybrid audit. We recalculated the risk on audit data from the closest race in Kalamazoo (Whitmer vs Schuette) using ALPHA with the optimized intersection supermartingale P-value P_M^*, ALPHA with the optimized Fisher P-value P_F^*, EB with P_F^*, and EB with P_M^*, and compared these with the SUITE P-value. Because we could not access the original order of sampled ballots in the ballot-polling stratum, we simulated P-values for 10,000 random ballot orders with the marginal totals in the sample. We computed the mean, standard deviation, and 90th percentile of these P-values for each method.

To get the ALPHA P-values, we used ALPHA-UB in the CVR stratum and ALPHA-ST in the no-CVR stratum. For EB P-values, we used the predictable mixture parameters of Waudby-Smith and Ramdas [22] to choose λ_{ik}, truncating at 0.75 in both strata. Sample allocation was dictated by the original pilot audit: 8 cards from the CVR stratum (5,294 votes cast; diluted margin 0.55) and 32 from the no CVR stratum (22,732 votes cast; diluted margin 0.57).

Table 3 presents P-values for each method. For ALPHA, the mean P_F^* is about half the SUITE P-value; for P_M^*, the mean is more than an order of magnitude smaller than the SUITE P-value. The P-value distributions for ALPHA are concentrated near the mean. On the other hand, the EB P_M^* and P_F^* P-values are both an order of magnitude larger than the SUITE P-value and their distributions are substantially more dispersed than the distributions of ALPHA P-values.

Table 3. Measured risks (P-values) computed from the 2018 Kalamazoo MI audit data. For SUITE, the original P-value is shown. For replications, the mean, standard deviation (SD), and 90th percentile of P-values in 10,000 reshufflings of the sampled ballot-polling data are shown.

Method	$\boldsymbol{P}$-value		
	Mean	SD	90th
SUITE	0.037	*	*
ALPHA P_F^*	0.018	0.002	0.019
ALPHA P_M^*	0.003	0.000	0.003
EB P_F^*	0.348	0.042	0.390
EB P_M^*	0.420	0.134	0.561

4.3 A Highly Stratified Audit

As mentioned in Sect. 3.6, many within-stratum risk-measuring functions do not yield tractable expressions for $P_F(\boldsymbol{\theta})$ or $P_M(\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$, making it hard to find the maximum P-value over the union unless K is small. Indeed, previous implementations of SUITE only work for $K = 2$. However, the combined log-P-value for EB is linear in $\boldsymbol{\theta}$ for P_M^* and piecewise linear for P_F^*. Maximizing the combined log-P-value over the union of intersections is then a linear program that can be solved efficiently even when K is large.

To demonstrate, we simulated a stratified ballot-polling audit of the 2020 presidential election in California, in which $N = 17,500,881$ ballots were cast across $K = 58$ counties (the strata), using a risk limit of 5%. The simulations assumed that the reported results were correct, and checked whether reported winner Joseph R. Biden really beat reported loser Donald J. Trump. The audit assumed that every ballot consisted of one card; workloads would be proportionately higher if the sample were drawn from a collection of cards that includes some cards that do not contain the contest. Sample sizes were set to be proportional to turnout, plus 10 cards, ensuring that at least 10 cards were sampled from every county. Risk was measured within strata by EB with predictable mixture λ_{ik} thresholded at 0.9 [22]. Within-stratum P-values were combined using P_F^* (P_M^* did not work well for EB with proportional allocation in simulations). To approximate the distribution of sample sizes needed to stop, we simulated 30 audits at each increment of 5,000 cards from 5,580 to 100,580 cards. We then simulated 300 audits at 70,580 cards, roughly the 90th percentile according to the smaller simulations.

In 91% of the 300 runs, the audit stopped by the time 70,580 cards had been drawn statewide. Drawing 70,580 ballots by our modified proportional allocation rule produces within-county sample sizes ranging from 13 (Alpine County, with the fewest voters) to 17,067 (Los Angeles County, with the most). A comparison or hybrid audit using sampling without replacement would presumably require inspecting substantially fewer ballots. It took about 3.5 s to compute each P-value in R (4.1.2) using a linear program solver from the `lpSolve` package (5.6.15) on a mid-range laptop (2021 Apple Macbook Pro).

5 Discussion

ALPHA intersection supermartingales were most efficient compared to the SUITE pilot audit in Michigan and in simulations. Lower-sided testing allocation was better than proportional allocation, especially for EB. Fisher pooling limits the damage that a poor allocation rule can do, but is less efficient than intersection supermartingales with a good stratum selection rule. For comparison audits, it helps to bet more aggressively than ALPHA-ST by using ALPHA-UB or EB. However, EB was not efficient compared to SUITE when replicating the Michigan hybrid audit due to poor performance in the ballot-polling stratum.

Our general recommendation for hybrid audits is: (i) use an intersection supermartingale test with (ii) adaptive stratum selection and (iii) ALPHA-UB (or another method that can exploit low sample variance to bet more aggressively) as the risk-measuring function in the comparison stratum and (iv) ALPHA-ST (or a method that "learns" the population mean) as the risk-measuring function in the ballot-polling stratum. When the number of strata is large, audits can leverage the log-linear form of the EB supermartingale to quickly find the maximum P-value, as illustrated by our simulated audit spread across California's 58 counties.

In future work, we hope to construct better stratum allocation rules and characterize (if not construct) optimal rules. The log-linear structure of the EB supermartingale may make it simpler to derive optimal allocation rules.

While stratum selection is not an instance of a traditional multi-armed bandit (MAB) problem, there are connections, and successful strategies for MAB might help. For instance, stratum selection could be probabilistic and involve continuous exploration and exploitation, in contrast to the "hard stop" rules we used in our simulations here.

A Computational details

The following describes details of the allocation simulations in Sect. 4. Within each stratum, we computed null means along an equispaced grid of $(2 \max\{N_1, N_2\})$ points[3] for $\theta_1 \in [\varepsilon_1, \theta/w_1 - \varepsilon_1]$ with $\theta_2 = (\theta - w_1\theta_1)/w_2$. The

[3] The cardinality was chosen so that a null mean was computed for every possible (discrete) value of θ_k. A finer grid is unnecessary; a coarser grid may not find the true minimum.

null means were then adjusted to $\beta_1 := \theta_1 + 1 - \bar{A}_1^c$ and $\beta_2 := \theta_1 + 1 - \bar{A}_2^c$. The conditional null means β_{i1} and β_{i2} were computed as:

$$\beta_{ik} = \frac{N_k \beta_k - \sum_{j=1}^{i-1} X_{ik}}{N_k - (i-1)}$$

Tuning parameters for ALPHA-ST were chosen as in Stark [19, Section 2.5.2] with $d_k = 20$ and the initial estimate τ_{0k} set to $u_k^A = 1$, the expected mean when there is no error in the CVRs. For ALPHA-UB, we set

$$\tau_{ik}^{\text{UB}} := \frac{(d_k \tau_{0k} + \sum_{j=1}^{i-1} X_{jk})/(d_k + i - 1) + f_k u_k / \hat{\sigma}_{ik}^2}{1 + f_k / \hat{\sigma}_{ik}^2}.$$

The first term in the numerator of τ_{ik}^{UB} is truncated shrinkage estimator ALPHA-ST. The second term biases τ_{ik}^{UB} towards u_k with a weight proportional to the inverse running sample variance $\hat{\sigma}_{ik}^2$. The constant of proportionality f_k is a tuning parameter set to $f_k := .01$; higher f_k would bias τ_{ik} towards u_k more aggressively. The variance-dependent bias amounts to betting more when the population variance is low, which it tends to be in comparison audits when the voting system works properly. Truncation keeps τ_{ik} within its allowed range.

For both ALPHA strategies, τ_{ik} was truncated to be in $[\beta_{ik} + \varepsilon_k, u_k(1-\delta)]$, where $\varepsilon_k := 1/2N_k$ was the minimum value of one assorter and $\delta = 2.220446 \times 10^{-16}$ was machine precision. If $\beta_{ik} + \varepsilon_k \geq u_k$, we set the corresponding terms in the supermartingale to 1: that (composite) null is true.

Each stratum selection rule was applied to every supermartingale. For proportional allocation, there was no additional selection: samples were gathered round-robin across strata, omitting any strata that were fully exhausted. For lower-sided testing, the sampling from a stratum ceased when the lower-sided test rejected at level .05. This was implemented by setting all future terms in the supermartingale equal to 1 after rejection. The stratumwise supermartingales were then multiplied to produce $2\max\{N_1, N_2\}$ intersection supermartingales and their minimum (over nulls) was found at each sample size. The reciprocal of this minimized intersection supermartingale was a sequence of P-values corresponding to P_M^* under a particular sample allocation rule. The same strategy, but using Fisher pooling, was used to find P_F^*. The sample size at risk limit $\alpha = 5\%$ is the sample size for which the P-value sequence first hits or crosses 0.05, summed across both strata.

B Data and Code

All code used in this paper is available at https://github.com/spertus/sweeter-than-SUITE. SUITE was applied to the Michigan RLA data in a Jupyter notebook available at https://github.com/kellieotto/mirla18. Reported results from California's 2020 presidential election are available at https://elections.cdn.sos.ca.gov/sov/2020-general/sov/csv-candidates.xlsx.

References

1. Appel, A.W. and Stark, P.B.: Evidence-based elections: Create a meaningful paper trail, then audit. Georgetown Law Technol. Rev. **4**(2), 523–541 (2020). https://georgetownlawtechreview.org/wp-content/uploads/2020/07/4.2-p523-541-Appel-Stark.pdf
2. Appel, A.W., DeMillo, R.A., Stark, P.B.: Ballot-marking devices cannot assure the will of the voters. Election Law J. Rules Polit. Policy **19**(3), 432–450 (2020). https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3375755
3. Baker, P., Haberman, M.: In torrent of falsehoods, trump claims election is being stolen. The New York Times, November 2020. ISSN 0362–4331. https://www.nytimes.com/2020/11/05/us/politics/trump-presidency.html
4. Chaitlin, D.: Sidney powell shares 270-page binder of documents buttressing election fraud claims, December 2020. https://www.washingtonexaminer.com/news/sidney-powell-shares-election-fraud-claims. Section: News
5. Hall, J., et al.: Implementing risk-limiting post-election audits in California. In: Proceedings of 2009 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE 2009), Montreal, Canada, August 2009. USENIX http://www.usenix.org/event/evtwote09/tech/full_papers/hall.pdf
6. Higgins, M., Rivest, R., Stark, P.: Sharper p-values for stratified post-election audits. Stat. Polit. Policy **2**(1) (2011). http://www.bepress.com/spp/vol2/iss1/7
7. Howard, L., Rivest, R., Stark, P.: A review of robust post-election audits: various methods of risk-limiting audits and Bayesian audits. Technical report, Brennan Center for Justice (2019). https://www.brennancenter.org/sites/default/files/2019-11/2019_011_RLA_Analysis_FINAL_0.pdf
8. Howard, S.R., Ramdas, A., McAuliffe, J., Sekhon, J.: Time-uniform, nonparametric, nonasymptotic confidence sequences. Ann. Stat. **49**(2) (2021). https://doi.org/10.1214/20-aos1991
9. Kahn, C.: Half of republicans say Biden won because of a 'Rigged' election: reuters/Ipsos poll. Reuters, November 2020. https://www.reuters.com/article/us-usa-election-poll-idUSKBN27Y1AJ
10. Levine, A.: Donald Trump's favorite voting machines, September 2020. http://washingtonmonthly.com/2020/09/23/donald-trumps-favorite-voting-machines/
11. Ottoboni, K., Stark, P.B., Lindeman, M., McBurnett, N.: Risk-limiting audits by stratified union-intersection tests of elections (SUITE). In: Krimmer, R., Volkamer, M., Cortier, V., Goré, R., Hapsara, M., Serdült, U., Duenas-Cid, D. (eds.) E-Vote-ID 2018. LNCS, vol. 11143, pp. 174–188. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00419-4_12
12. Pesarin, F., Salmaso, L.: Permutation Tests for Complex Data: Theory, Applications, and Software. Wiley, West Sussex (2010)
13. Stark, P.: Conservative statistical post-election audits. Ann. Appl. Stat. **2**, 550–581 (2008). http://arxiv.org/abs/0807.4005
14. Stark, P.: A sharper discrepancy measure for post-election audits. Ann. Appl. Stat. **2**, 982–985 (2008). http://arxiv.org/abs/0811.1697
15. Stark, P.: CAST: canvass audits by sampling and testing. IEEE Trans. Inf. Forensics Secur. Spec. Issue Electron. Voting **4**, 708–717 (2009)
16. Stark, P.: Auditing a collection of races simultaneously. Technical report. arXiv.org (2009). http://arxiv.org/abs/0905.1422v1
17. Stark, P.: Delayed stratification for timely risk-limiting audits. https://www.stat.berkeley.edu/~stark/Preprints/delayed19.pdf (2019)

18. Stark, P.B.: Sets of half-average nulls generate risk-limiting audits: SHANGRLA. In: Bernhard, M., et al. (eds.) FC 2020. LNCS, vol. 12063, pp. 319–336. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-54455-3_23
19. Stark, P.: ALPHA: audit that learns from previously hand-audited ballots. Annals of Applied Statistics, Conditionally accepted 2022. https://arxiv.org/abs/2201.02707
20. Stark, P., Wagner, D.: Evidence-based elections. IEEE Secur. Priv. **10**, 33–41 (2012). https://www.stat.berkeley.edu/~stark/Preprints/evidenceVote12.pdf
21. Ville, J.: Étude critique de la notion de collectif (1939). http://eudml.org/doc/192893
22. Waudby-Smith, I., Ramdas, A.: Estimating means of bounded random variables by betting (2020). https://arxiv.org/abs/2010.09686
23. Waudby-Smith, I., Stark, P.B., Ramdas, A.: RiLACS: risk limiting audits via confidence sequences. In: Krimmer, R., et al. (eds.) E-Vote-ID 2021. LNCS, vol. 12900, pp. 124–139. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86942-7_9

They May Look and Look, Yet Not See: BMDs Cannot be Tested Adequately

Philip B. Stark[1](✉) and Ran Xie[2]

[1] University of California, Berkeley, CA, USA
stark@stat.berkeley.edu
[2] Stanford University, Stanford, CA, USA

Abstract. Bugs, misconfiguration, and malware can cause ballot-marking devices (BMDs) to print incorrect votes. Several approaches to testing BMDs have been proposed. In *logic and accuracy testing* (LAT) and *parallel* or *live* testing, auditors input known test votes into the BMD and check whether the printout matches. *Passive* testing monitors the rate at which voters "spoil" BMD printout, on the theory that if BMDs malfunction, the rate will increase noticeably. We provide lower bounds that show that these approaches cannot reliably detect outcome-altering problems, because: (i) The number of possible voter interactions with BMDs is enormous, so testing interactions uniformly at random is hopeless. (ii) To probe the space of interactions intelligently requires an accurate model of voter behavior, but because the space of interactions is so large, building a sufficiently accurate model requires observing an enormous number of voters in every jurisdiction in every election—more voters than there are in most U.S. jurisdictions. (iii) Even with a perfect model of voter behavior, the required number of tests exceeds the number of voters in most U.S. jurisdictions. (iv) An attacker can target interactions that are intrinsically expensive to test, e.g., because they involve voting slowly; or interactions for which tampering is less likely to be noticed, e.g., because the voter uses the audio interface. (v) Whether BMDs misbehave or not, the distribution of spoiled ballots is unknown and varies by election and possibly by ballot style: historical data do not help much. Hence, there is no way to calibrate a threshold for passive testing, e.g., to guarantee at least a 95% chance of noticing that 5% of the votes were altered, with at most a 5% false alarm rate. (vi) Even if the distribution of spoiled ballots were known to be Poisson, the vast majority of jurisdictions do not have enough voters for passive testing to have a large chance of detecting problems but only a small chance of false alarms.

Keywords: Logic and accuracy testing · Parallel testing · Live testing

1 Introduction

BMDs print votes, often as barcodes or QR codes, together with a human-readable text summary (some BMD printout resembles a hand-marked paper

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 122–138, 2022.
https://doi.org/10.1007/978-3-031-15911-4_8

ballot, HMPB). Jurisdictions including the U.S. state of Georgia, Los Angeles County, California, and Philadelphia, Pennsylvania, recently purchased BMDs for all in-person voters to use.

Bugs, misconfiguration, or malware can make the printed votes and QR codes differ from each other and from the voter's selections. Some have argued that this does not compromise election integrity because voters have the opportunity to inspect BMD printout and to start over if the printout does not match their intended selections; and that since voters can make mistakes hand-marking ballots, HMPBs are no more secure or reliable than BMD printout [19]. We find those arguments unpersuasive:

- In some jurisdictions, the official record of the vote for counts and recounts is the QR code, which voters cannot check.[1]
- The arguments equate holding voters responsible for their own errors with holding voters responsible for the overall security of the system [1,2].
- Most voters *do not* inspect BMD printout [3,5,10]. Those who do rarely detect actual errors [3,13]. To reliably detect errors entails voters taking 3–6 *minutes* to compare a written slate of candidates with the printed selections [12], but voters generally spend less than 3 *seconds* reviewing BMD printout [5,10].
- If a BMD misprints a voter's selections, only the voter can get evidence of the problem: elections conducted using BMDs are not *contestible* [1].
- If BMDs misbehave, there is no way to determine the correct election outcome because there is no trustworthy paper record of the vote: BMDs are not *strongly software independent* [21].

Concerns about BMDs are not merely hypothetical: BMDs have caused scanners to fail to count votes accurately, to allow voters to vote, and to present all voting options to voters, even after passing LAT [4,6,9,15,18,20,23,24,30,33].

BMD advocates also claim BMDs eliminate ambiguous marks, prevent overvotes, and warn about undervotes (e.g., [19]). But that presumes BMDs function correctly; the rate of truly ambiguous handmade marks is minuscule [1]; and precinct-based optical scanners also protect against undervotes and overvotes (the Voluntary Voting Systems Guidelines, VVSG, require it).[2] Regardless, elections conducted using BMDs are not trustworthy unless there is a way to ensure that BMD misbehavior did not change any outcome. (If the paper trail itself is not trustworthy, risk-limiting audit procedures do not help because even an accurate full hand count may not reveal who really won.) Elections—and hence BMDs—need to be protected against malicious, technically capable attackers, such as nation states.[3] If testing has a high chance of detecting that an outcome

[1] See https://rules.sos.ga.gov/gac/183-1-15-.03?urlRedirected=yes&data=admin&lookingfor=183-1-15-.03 (last visited 5 May 2022). Audits in Georgia rely on the human-readable text, but legally cannot correct outcomes, and are conducted only for one contest every two years.

[2] Such protection has been required since VVSG 1.0; see Sect. 2.3.3.2 of [27].

[3] The U.S. Senate Intelligence Committee, the Department of Homeland Security, and the FBI concluded that Russian state hackers attacked U.S. elections in 2016 [31].

was altered by a skilled attacker, it also protects against misconfiguration and bugs—which attackers could mimic.

2 Prior Work

Vulnerabilities of particular BMDs are discussed in depth in expert declarations by J. Alex Halderman in *Curling et al. v. Raffensperger et al.*. Theoretical vulnerabilities of various BMD designs are discussed in [1]. [7,32] discuss testing BMDs; here, we quantitatively investigate their heuristic claims. Three approaches to testing BMDs have been proposed: pre-election logic and accuracy testing (LAT), "live" or "parallel" testing during the election, and "passive" testing by monitoring the spoiled ballot rate. In LAT and parallel testing, auditors make selections on a BMD then check whether the printout accurately reflects those selections. The primary difference is that LAT happens before the election and parallel testing happens during the election. *Passive* testing uses the spoiled ballot rate: if more voters than usual request a do-over, that might be because the machines are malfunctioning.

3 How Much Testing is Enough?

If the paper trail accurately reflects who won, accurate full hand counts and risk-limiting audits (RLAs) can catch and correct wrong outcomes. Here, we study whether testing can establish with high confidence that a paper trail printed by BMDs accurately reflects who won. If not, a recount need not show who really won, and a genuine RLA is impossible.

3.1 Threats and Defenses

We make the following assumptions about BMD threats and defenses:

1. Attackers seek to alter the outcome of one or more contests without being detected. (Some might want to be detected, to undermine public confidence.)
2. Attackers know the testing strategy. This does not preclude the possibility that the strategy will be adaptive or have a random element.
3. Attackers have access to the state history of each BMD, including votes, machine settings, etc.; auditors do not.
4. Attackers have an accurate model of voter behavior in past elections, including political preferences, voting speed, BMD settings, and so on; auditors generally do not, because it would require monitoring voters illegally.
5. Auditors seek to ensure that if any outcome is altered, there is a high chance of detecting it, while keeping the chance of false alarms small.
6. Auditors do not know which contest(s), if any, were altered.
7. Auditors must obey the law and protect voter privacy.

3.2 Jurisdiction Sizes, Contest Sizes, and Margins

U.S. elections are typically administered by counties, townships, or other political units smaller than states. A *ballot style* corresponds to the collection of contests a given voter is eligible to vote in. Typically in the U.S., some contests are on only a fraction of ballot styles in a jurisdiction, in part because many small political units have elections for various offices and measures. Many contests of all sizes are decided by small margins. For instance, in Georgia, U.S., the reported margin in the 2020 presidential election was about 0.2%.

Few votes need to be changed to alter the outcome of small contests and contests with small margins. Conversely, the number of voters in a jurisdiction is an upper bound on the number of passive tests that can be performed and on the sample size to "learn" voter behavior for efficient parallel testing. Thus, jurisdiction size is an important constraint on BMD testing. Since ballot layout, contests, equipment, demographics, political preferences, and other variables vary across and within jurisdictions and malware could affect only some equipment or ballot styles, it is not possible to pool data across jurisdictions to get more power.

Changing votes on 1% of ballots in a jurisdiction can alter the margin of a jurisdiction-wide plurality contest by 2% if there are no undervotes or invalid votes in that contest. If the undervote rate is 30%, then changing votes on 1% of the ballots can change the margin by $0.02/0.7 = 2.9\%$. If a contest is only on 10% of the ballots and the undervote rate in the contest is 30%, altering the votes on 1% of ballots could change the margin in that contest by nearly 29%.

As of 2020, only 1,629 U.S. cities had populations of 100,000 or more, of over 81,363 incorporated places [26]. The 2020 median population of U.S. incorporated areas is 1,201, so about half of the 81,363 incorporated places have turnout of 1,201 or fewer voters. Thus, an attacker does not have to change many votes to alter the outcome of a typical contest for an elected official in a U.S. city or incorporated township. According to [29] the 2020 median turnout in the 6,405 U.S. counties with recorded active voter data was 4,470 voters, and turnout was less than 11,500 voters for more than 2/3 of jurisdictions. In 65.5% of states, more than 50% of counties have fewer than 30,000 active voters. In 85.5% of states, more than 50% of counties have fewer than 100,000 active voters.

3.3 Voting Transactions

We shall call a voter's interaction with a BMD a *voting transaction* or *transaction* (see Table 1). Transactions are characterized by many variables, including:

- when the transaction starts
- time since the previous voter finished (a measure of polling-place congestion)
- number of transactions before the current transaction
- the voter's sequence of selections and revisions of selections
- the time to make each selection before taking another action
- whether the voter looks at every page of options in each contest
- the time the voter spends reviewing and revising selections

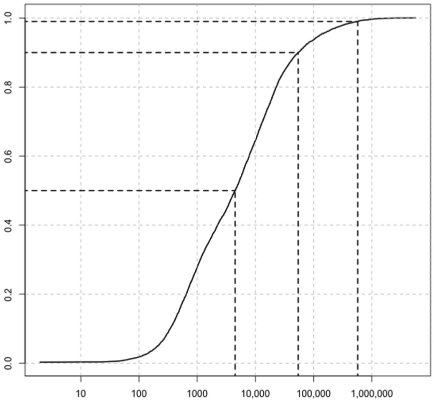


Fig. 1. 2020 turnout by jurisdiction in 3073 counties [29]. Turnout was below 10,000 in ≈50% of counties.

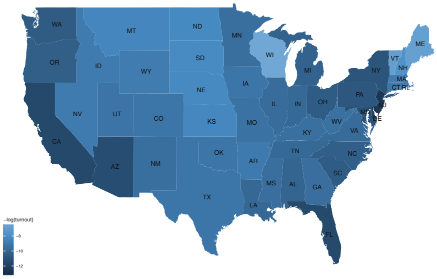


Fig. 2. Median 2020 turnout by jurisdiction in the U.S. [29]

– precisely where voter touches the screen
– BMD settings, including font size, language, use of audio, volume, tempo, pausing, rewinding, use of the sip-and-puff interface, inactivity warnings

Table 1 lists some of the variables and the number of values they can take.[4] The huge number of possible transactions helps an attacker pick a subset large enough to change an outcome but that auditors are unlikely to probe.

[4] Table 1 assumes all contests are "vote-for-one." Ranked-choice voting and multi-winner plurality contests yield more possible transactions. Continuous variables were binned into a few options. VVSG 1.1 [28] requires: (a) Alternative language access is mandated under the Voting Rights Act of 1975, subject to certain thresholds (e.g., if the language group exceeds 5% of the voting age population). (b) The voting system shall provide the voter the opportunity to correct the ballot for either an undervote or overvote before the ballot is cast and counted. (c) An Acc-VS with a color electronic image display shall allow the voter to adjust the color saturation throughout the transaction while preserving the current votes. (d) At a minimum, two alternative display options listed shall be available: 1) black text on white background, 2) white text on black background, 3) yellow text on a black background, or 4) light cyan text on a black background. (e) A voting system that uses an electronic image display shall be capable of showing all information in at least two font sizes. (f) The audio system shall allow the voter to control the volume throughout the voting transaction while preserving the current votes. (g) The volume shall be adjustable from a minimum of 20dB SPL up to a maximum of 100 dB SPL, in increments no greater than 10 dB. (h) The audio system shall allow the voter to control the rate of speech throughout the voting transaction while preserving the current votes. (i) The range of speeds supported shall include 75% to 200% of the nominal rate.

Table 1. Some parameters of BMD transactions and their number of possible values.

Parameter	Optimistic	More realistic
Contests	3	20
Candidates per contest	2	4
Languages	2	13
Time of day	10	20
Number of previous voters	5	140
Undervotes	2^3	2^{20}
Changed selections	2^3	2^{20}
Review	2	2
Time per selection	2	5^{20}
Contrast/saturation	–	4
Font Size	2	4
Audio Use	2	2
Audio tempo	–	4
Volume	5	10
Audio pause	-	2^{20}
Audio + video	–	2
Inactivity warning	2	2^{20}
Total combinations	6.14×10^6	3.4×10^{48}

4 Passive Testing

Passive testing sounds an alarm if the number of spoiled ballots exceeds some threshold, t. To ensure that passive testing has a false negative rate (failing to detect altered outcomes) of at most $X\%$, we need to know that the chance that the number of spoiled ballots is greater than or equal to t is at least $X\%$ if BMDs altered any outcome. Conversely, to limit the false alarm rate to at most $Y\%$, we need to know that the chance that the number of spoiled ballots is greater than or equal to t is at most $Y\%$ if BMDs function correctly.

Finding such a value of t is impossible in practice because the distribution of spoiled ballots may depend on ballot design, voting rules, the number of contests, and other things that vary from election to election and place to place—and when BMDs misbehave, also on the number of altered transactions, and the voters and contests affected. Hence, to lower-bound the difficulty, we will assume (optimistically) that the number of spoiled ballots has a Poisson distribution whether BMDs behave correctly or not; but with a rate that depends on the rate of altered transactions. We assume either that 7% of voters will notice errors and spoil their ballots, consistent with the findings of [3], or that 25% of voters will. We consider contest margins of 1%–5% and rates of false positives (false alarms) and false negatives (failing to notice altered outcomes) of 5% and 1%. Results are in Table 2; software to calculate these numbers is in https://github.com/pbstark/Parallel19.

Combining Table 2 and Fig. 1 shows that even if the probability distribution of spoiled ballots were known to be Poisson and the spoilage rate when equipment functions correctly were known perfectly, in 2020, in 58.2% of U.S. states fewer than half the counties had enough voters for passive testing to work, even in county-wide contests, on the assumption that 7% of voters whose votes are altered will spoil their ballots.

If turnout is roughly 50%, jurisdiction-wide contests in jurisdictions with fewer than 60,000 voters—22 of California's 58 counties in 2020 [29]—cannot in

Table 2. Minimum turnout for passive testing with a 5% false negative rate to have at most a 5% false positive rate (cols 3–5) or for passive testing with a 1% false negative rate to have at most a 1% false positive rate (cols 6–8), as a function of the the contest margin (col 1), the percentage of voters who would notice errors (col 2), and the base rate at which voters spoil BMD printout. The number of spoiled ballots is assumed to have a Poisson distribution, with known rate, absent malfunctions. Malfunctions increase the rate by half the margin times the detection rate.

Margin	Voter detection rate	5% error rate			1% error rate		
		Base spoilage rate			Base spoilage rate		
		0.5%	1%	1.5%	0.5%	1%	1.5%
1%	7%	451,411	893,176	1,334,897	908,590	1,792,330	2,675,912
	25%	37,334	71,911	106,627	76,077	145,501	214,845
2%	7%	115,150	225,706	336,160	233,261	454,295	675,242
	25%	9,919	18,667	27,325	20,624	38,039	55,442
3%	7%	52,310	101,382	150,471	106,411	204,651	302,864
	25%	4,651	8,588	12,445	9,870	17,674	25,359
4%	7%	30,000	57,575	85,227	61,385	116,631	171,908
	25%	2,788	4,960	7,144	5,971	10,312	14,681
5%	7%	19,573	37,245	54,932	40,156	75,671	110,989
	25%	1,838	3,274	4,689	4,036	6,849	9,650

principle limit the chances of false positives and false negatives to 5% for margins below 4%, even under these optimistic assumptions. For contests that involve only part of a jurisdiction, the situation is worse.

4.1 Targeting Vulnerable Voters

The analysis above assumes that all voters are equally likely to detect errors and spoil their ballots. But an attacker can use BMD settings, state history, and session data to target voters who are less likely to notice problems.

Voters with Visual Impairments. Approximately 0.8% of the U.S. population is legally blind; approximately 2% age 16 to 64 have a visual impairment [16]. Current BMDs do not provide voters with visual impairments a way to check the printout. If an attacker only alters votes when the voter uses the audio interface or large fonts, detection may be very unlikely.

Voters with Motor Impairments. Some BMDs allow voters to print and cast a ballot without looking at it, for instance the ES&S ExpressVote® with "Autocast," aka "permission to cheat" [1]. The attacker can change every vote cast using Autocast, with zero chance of detection.

Voters who use Languages other than English. U.S. law requires some jurisdictions to provide ballots in languages other than English. For instance, Los Angeles County, CA, provides voting materials in 13 languages [14]. In 2013, roughly 26% of voters in Los Angeles County spoke a language other than English at home [14]. It is our understanding that BMDs generally print only in English. If voters who use a foreign language on the BMD are unlikely to check the English-language printout, an attacker could change the outcome of contests with large margins with little chance of detection.

Fast and Slow Voters. An attacker can monitor how long it takes voters to make their selections, whether they change selections, how long they review the summary screen, etc. A voter who spends little time reviewing selections onscreen may be unlikely to review the printout carefully. Conversely, a voter who takes a very long time to make selections or changes selections repeatedly might find voting difficult or confusing and be unlikely to notice errors.

Thus, it is in the attacker's interest to target the same groups of voters BMDs are supposed to help: voters with visual impairments, voters with limited dexterity, voters who use a language other than English, and voters with cognitive disabilities.

4.2 FUD Attacks on Passive testing

Even under ideal circumstances, passive testing does not produce direct evidence of problems; it does not identify which ballots or contests have errors; and it does not provide any evidence about whether problems changed outcomes. Relying on spoiled ballots as a sign of fraud opens the door to a simple, legal way to undermine elections: encourage voters to spoil ballots.

5 LAT and Parallel Testing

Suppose that malware alters one or more votes with probability p, independently across transactions, uniformly across voters—regardless of the voter's selections or any aspect of the transaction that the attacker can ascertain. Then if auditors make n tests, the chance that the BMD will alter at least one of the votes in at least one of the tests—and the attack will be detected—is $1 - (1 - p)^n$. For $p = 0.01$, $n = 300$ tests would give a 95% chance of detecting a problem.

A BMD can handle roughly 140 transactions per day. Testing enough to have a 95% chance of detecting a 1% problem on one BMD would leave no time for voters to use that BMD. Even for pre-election LAT, where capacity for actual voters is not an issue, conducting 300 "typical" tests would take about 25 h.

If there were a large number of machines known to have been (mis)programmed identically, tests could be spread across them. But there are many small contests that need to be tested in conjunction with all other contests that appear on any ballot style that contains them, and there is no guarantee that all BMDs in a jurisdiction are programmed identically.

This threat model is completely unrealistic. An attacker who wants Alice to beat Bob will not alter votes for Alice: it would needlessly increase the chance of detection. And as discussed in Sect. 4.1, rather than randomly changing votes for Bob into votes for Alice, an attacker can target transactions that auditors are unlikely to probe.

Setting aside specific machines for testing facilitates a "Dieselgate" type attack [11], as does conducting tests on a schedule, as suggested by [7]. Tests need to be unpredictable—with respect to the specific BMDs tested, time, vote pattern, duration, and other characteristics of voting transactions—or attackers can avoid detection by altering only transactions that do not correspond to any test. There may be pressure to reduce testing when BMDs are busy, to reduce waiting times. Because malware can monitor the pace of voting, reducing testing when machines are busy makes it easier to avoid detection.

An attacker need not alter many transactions to change the outcome of small contests and contests with small margins. The fewer votes altered, the more tests required to ensure a large chance of detection. To test efficiently, tests should sample more common transactions with higher probability. Attackers might be able to estimate of the distribution of transactions using malware installed on BMDs in previous elections, but testers will not, since it involves tracking voter behavior at a level of detail that violates voter privacy. See assumptions 4 and 7 and Sect. 5.2.

Auditors do not know which contest(s) and candidate(s) are affected. To have a large chance of detecting interference, there needs to be a large chance of testing a transaction the attacker alters. Attackers can target transactions that are intrinsically expensive to test, e.g., transactions that take longer than 10 min, transactions in which the voter changes some number of selections, transactions that display the ballot in a language other than English, transactions that use the audio interface at a reduced tempo, etc.

5.1 Lower Bounds on the Difficulty of Parallel Testing

We now study an idealized version of parallel testing, where auditors can tell whether a random sample of BMD printouts accurately show the voters' selections. Suppose a contest has 4,470 voters, the median jurisdiction turnout in 2020. Suppose that malware alters votes in 23 transactions, which could change a margin by more than 1% in a jurisdiction-wide contest. How many randomly selected printouts would need to be checked to have at least a 95% chance of finding at least one with an error? The answer is the smallest n such that

$$\frac{4470-23}{4470}\cdot\frac{4469-23}{4469}\cdots\frac{4470-(n-1)-23}{4470-(n-1)}\leq 0.05, \tag{1}$$

i.e., $n = 546$ printouts, about 12.2% of the transactions, corresponding to testing each BMD several times per hour.

Conversely, suppose auditors randomly check 13 printouts per day per machine (on average, testing hourly for a 13-hour day, ≈9.2% of BMD capacity). To have at least a 95% chance of detecting that the outcome of a contest with a 1% margin was altered, there would need to be at least 6,580 voters in the contest (almost 150% the median turnout in jurisdictions across the U.S.), corresponding to 47 BMDs, even under these unrealistically optimistic assumptions.

5.2 Building a Model of Voter behavior

In practice, auditors cannot check whether voters' BMD printout is correct. Instead of sampling voters' actual transactions in the election, they will have to come up with their own test transactions. Testing transactions uniformly at random from all possible transactions is doomed because the number of possible transactions is so large. To mimic voters, auditors might consider sampling from P, the population distribution of voting transactions, i.e., the fraction of voters who use the BMD in each of the $S = 6.14 \times 10^6$ ways in the optimistic estimate in Table 1. Suppose an attacker wants to change the outcome of a contest with a margin of m, expressed as a fraction of ballots cast (rather than as a number of votes). The attacker only needs to change a fraction $m/2$ of the transactions to change the margin by m. To have probability at least $1-\alpha$ of detecting a change to the outcome of any contest with true margin m, auditors must test in a way that has probability at least $1-\alpha$ of sampling at least once from every subset of transactions that contains a fraction $m/2$ of the transactions. If auditors could sample transactions independently at random from P, each sample transaction would have probability $1 - m/2$ of *not* being one of the altered transactions. The chance that t randomly selected transactions would not include one that is altered would be $(1 - m/2)^t$. Thus the number of transactions auditors would need to test is the smallest t for which

$$(1 - m/2)^t \le \alpha, \quad \text{i.e.,} \tag{2}$$

$$t \ge \frac{\log \alpha}{\log(1 - m/2)}. \tag{3}$$

This is essentially Eq. 1 for sampling with replacement; the two are indistinguishable when t is small compared to the total number of possible transactions. A key difference is that in Eq. 1, auditors are sampling from the actual transactions in the election, while in Eq. 3, auditors are sampling from a model, the frequency distribution of of transactions.

In practice, the auditors do not know P—they will have to estimate it by monitoring voters. In reality, this is impossible to do well: (i) In a given election, P will depend on the particular contests on the ballot and the particular voters who participate, both of which change from election to election. (ii) The variables that characterize a voting transaction include the voter's selections and details about how the voter uses the BMD, so collecting the data would violate voter privacy illegally. To get a sense of the *statistical* difficulty of the problem, we ignore these practical difficulties. If auditors could select voters at random (with replacement) and observe in detail how they use the BMD—all the variables in Table 1—that would yield independent, identically distributed (IID) draws from P, which could be used to make an estimate, $\hat{P}$. If $\hat{P}$ differs too much from P, no number of tests will suffice, because $\hat{P}$ might estimate that the frequency of a transaction is zero when in fact it is sufficiently frequent that altering it could change an outcome. (By assumption 4, above, the attacker knows P and hence can exploit differences between $\hat{P}$ and P.) How many voters would auditors have to observed to ensure (with sufficiently high probability) that $\hat{P}$ is accurate enough for parallel testing?

The L_1 distance between two distributions bounds the difference in the probability they assign to any set ($|\hat{P}(A) - P(A)| \le \|\hat{P} - P\|_1/2$). If $\|\hat{P} - P\|_1 \ge m$, there may be a set A of transactions for which $P(A) = m/2$ but $\hat{P}(A) = 0$, so changing votes for transactions in A could alter some margin[5] by m, with zero chance of detection, no matter how many tests are performed, if the tests are drawn from $\hat{P}$ rather than P.

We cannot guarantee that $\|\hat{P} - P\|_1 \leq \varepsilon$ with *certainty*, but by observing enough randomly selected voters, we can ensure that the chance that $\|\hat{P} - P\|_1 > \varepsilon$ is at most β. If $\alpha \leq \beta$, even an infinite number of tests drawn from $\hat{P}$ may not suffice to guarantee chance at least $1 - \alpha$ of detecting outcome-changing manipulations. If $\alpha > \beta$, to guarantee chance at least $1 - \alpha$ of catching an outcome-changing error, the minimum number of tests required is

$$\min \left\{ t : (1 + \varepsilon/2 - m/2)^t \leq \frac{\alpha - \beta}{1 - \beta} \right\}. \tag{4}$$

Minimax Lower Bounds. Suppose auditors draw an IID sample of n transactions from P, a frequency distribution on S possible transactions. Let $\mathcal{M}_S$ denote the collection of all frequency distributions for those transactions. Then the training sample size n must be at least large enough to ensure that the L_1 error of the best estimator $\hat{P}$ is unlikely to exceed ε, provided $P \in \mathcal{M}_S$:

$$\inf_{\hat{P}} \sup_{P \in \mathcal{M}_S} \Pr_P \{ \|\hat{P} - P\|_1 \leq \varepsilon \} \geq 1 - \beta. \tag{5}$$

Theorem. ([8]) For any $\zeta \in (0, 1]$,

$$\begin{aligned} \inf_{\hat{P}} \sup_{P \in \mathcal{M}_S} \mathbb{E}_P \|\hat{P} - P\|_1 \geq \frac{1}{8} \sqrt{\frac{eS}{(1+\zeta)n}} \mathbb{1} \left(\frac{(1+\zeta)n}{S} > \frac{e}{16} \right) \\ + \exp\left(-\frac{2(1+\zeta)n}{S} \right) \mathbb{1} \left(\frac{(1+\zeta)n}{S} \leq \frac{e}{16} \right) \\ - \exp\left(-\frac{\zeta^2 n}{24} \right) - 12 \exp\left(-\frac{\zeta^2 S}{32 (\ln S)^2} \right), \end{aligned} \tag{6}$$

where the infimum is over all $\mathcal{M}_S$-measurable estimators $\hat{P}$.

Lemma. Let X be a random variable with variance $\mathbb{V}arX \leq 1$, and let $\beta \in (0, 1)$. If $\Pr\{X \geq \mathbb{E}X + \lambda\} \leq \beta$ then $\lambda \geq -\sqrt{\beta/(1-\beta)}$.

Proof. Suppose $\lambda \geq 0$. Then $\lambda \geq -\sqrt{\beta/(1-\beta)}$. Suppose $\lambda < 0$. By Cantelli's inequality and the premise of the lemma,

$$\beta \geq \Pr\{X \geq \mathbb{E}X + \lambda\} \geq 1 - \frac{\sigma^2}{\sigma^2 + \lambda^2} = \frac{\lambda^2}{\sigma^2 + \lambda^2} \geq \frac{\lambda^2}{1 + \lambda^2}. \tag{7}$$

[5] The set of undetectable shifts of $m/2$ votes might not include the one that any particular attacker seeks; this bound is worst-case across hypothetical attackers and distributions of transactions.

Solving for λ yields the desired inequality. $\square$.

Now $0 \leq \|\hat{P} - P\|_1 \leq 2$, so $\mathrm{V}ar\|\hat{P} - P\|_1 \leq 1$. By the lemma, we need $\lambda \geq -\sqrt{\beta/(1-\beta)}$ to ensure that $\Pr\{\|\hat{P}-P\|_1 \geq \mathbb{E}X + \lambda\} \leq \beta$. If $\|\hat{P}-P\|_1 \geq 2r$, there can be a set of transactions τ such that $P(\tau) = m/2$ but $\hat{P}(\tau) = 0$, so if tests are generated randomly according to $\hat{P}$ there is zero probability of testing any transaction in τ, no matter how many tests are performed. Thus if $\Pr\{\|\hat{P} - P\|_1 \geq m\} > \alpha$, even an infinite number of tests cannot guarantee chance at least $1 - \alpha$ detecting that a fraction $m/2$ of the transactions were altered, enough to wipe out a margin of m. By the lemma, that is the case if $m < \mathbb{E}\|\hat{P} - P\|_1 - \sqrt{\alpha/(1-\alpha)}$, i.e., if $\mathbb{E}\|\hat{P} - P\|_1 > m + \sqrt{\alpha/(1-\alpha)}$.

The theorem gives a family of lower bounds on $\mathbb{E}\|\hat{P} - P\|_1$ in terms of n. If the lower bound exceeds $m + \sqrt{\alpha/(1-\alpha)}$, testing by drawing transactions from $\hat{P}$ cannot protect against all outcome-changing errors. The bound grows with S, the number of possible transactions. To be optimistic, we use an unrealistically small value $S = 6.14 \times 10^6$ (Table 3).[6]

To guarantee a 95% chance of detecting that $m/2 = 5\%$ of transactions were altered, which could change jurisdiction-wide margins by 10% or more, the training sample would need to include at least 1.082 million transactions, even if auditors could conduct an infinite number of parallel tests. That is larger than the turnout in 99.7% of U.S. jurisdictions in 2020 [29]; it is roughly 0.5% of the U.S. voting population. To guarantee 99% chance of detecting that 0.5% of transactions were altered, which could change jurisdiction-wide margins by 1% or more, would require observing 3.876 million voters in complete, privacy-eliminating detail—more than the turnout in 99.9% of U.S. jurisdictions in 2020 [29], roughly 1.9% of the U.S. voting population.

Table 3. Lower bound on the sample size (col 4) required to estimate the distribution of voting transactions well enough to ensure the probability (col 1) of detecting the manipulation of the fraction of transactions (col 3) using some number of tests (col 2), if the support of the distribution of transactions has $S = 6.14 \times 10^6$ points.

Confidence level	Maximum tests	Altered votes	Bound (millions)
99%	2000	0.5%	3.87
		1%	3.58
		3%	2.69
		5%	2.09
95%	2000	0.5%	1.67
		1%	1.59
		3%	1.31
		5%	1.10
99%	Inf	0.5%	3.73
		1%	3.46
		3%	2.61
		5%	2.04
95%	Inf	0.5%	1.65
		1%	1.57
		3%	1.29
		5%	1.08

[6] Software implementing the calculations is in https://github.com/pbstark/Parallel19.

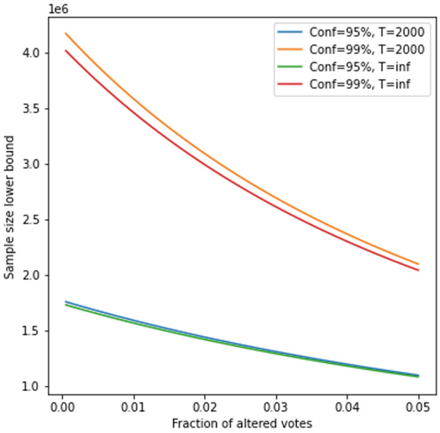


Fig. 3. Minimum training sample sizes as a function of the fraction of altered votes.

6 Complications

Reality is worse than the optimistic assumptions in our analyses:

Margins are not Known in Advance. Margins are not known until the election is over, when it is too late to do more testing if contests have narrower margins than anticipated. Testing to any pre-defined threshold, e.g., a 95% chance of detecting changes to 0.5% of the votes in any contest, will not always suffice.

Tests have Uncertainty. If the BMD printout reflects the wrong electoral outcome, a perfect full manual tally, recount, or risk-limiting audit based on BMD printout will confirm that wrong outcome. Suppose one could design practical parallel tests that had a 95% chance of sounding an alarm if BMDs alter $\geq 0.5\%$ of the votes in any contest. A reported margin of 1% or less in a plurality contest is below the "limit of detection" of such tests. Would laws require a runoff whenever a reported margin is below the limit of detection of the tests?

Special Risks for Some Voters. As discussed above in Sect. 4.1, BMDs can be used to selectively disenfranchise voters with disabilities and voters whose preferred language is not English. Indeed, the attacker's best strategy is to target such voters, in part because poll workers might more likely to think that complaints by such voters reflect voter mistakes rather than BMD malfunctions.

The only Remedy is a New Election. If a BMD is caught misbehaving, it should be removed from service and all BMDs in that jurisdiction should be investigated. But there is no way to determine the correct outcome or which votes were affected: BMDs are not *strongly software independent* [21].

7 Conclusion

We show that to protect against outcome-altering BMD malfunctions requires orders of magnitude more testing than is feasible. To our knowledge, no jurisdiction has conducted *any* parallel testing of BMDs of the kind suggested by [7,32], much less enough to reliably detect outcome-changing errors, bugs, or hacks.

Even if it were possible to test enough to get high confidence that no more than some threshold percentage the votes were changed in any contest, fairness would demand a runoff in contests decided by less than that threshold.

Some BMDs may be the best extant technology for voters with particular disabilities to mark and cast a paper ballot independently. But many BMDs are poorly designed. Some have easily exploited security flaws [1] and some do not enable voters with common disabilities to vote independently [22, pp. 68–90]. To our knowledge, no VVSG-certified BMD system provides a means for blind voters to check whether the printout matches their intended selections.

Using BMDs makes elections less trustworthy, less resilient, less transparent, more fragile, and more expensive [1,17]. BMDs have failure modes that hand-marked paper ballots do not have, and lack resilience when failures occur [1]. BMDs shift the burden of ensuring that voting equipment functions correctly from officials to voters, but do not provide voters any way to prove that they observed problems, if they do; nor can election officials show that outcomes are correct despite any problems that might have occurred [1]. BMDs undermine the ability of election officials to provide affirmative evidence that outcomes are correct, the fundamental principle of "evidence-based elections" [2,25].

Voters who use BMDs should be urged to bring a written list of their selections to the polls to check against BMD printout, and to request a fresh ballot if the printout does not match their intended selections. Election officials should track spoiled BMD printouts. There should be research on how to encourage voters to check BMD printout and report discrepancies, how to ensure the checks are accurate, and how to ensure that any reported problems are accountably and transparently recorded, addressed, and publicized; these issues also arise in end-to-end cryptographically verifiable (E2E-V) voting systems. For the foreseeable future, prudent election administration requires keeping the use of BMDs to a minimum.

Acknowledgements. We are grateful to Yanjun Han and Tsachy Weissman for helpful conversations about minimax L_1 estimation, and to Peter Rønne and anonymous referees for helpful comments and suggestions.

References

1. Appel, A., DeMillo, R., Stark, P.: Ballot-marking devices (BMDs) cannot assure the will of the voters. Election Law J. **19**, 432–450 (2020). https://doi.org/10.1089/elj.2019.0619
2. Appel, A., Stark, P.: Evidence-based elections: create a meaningful paper trail, then audit. Georgetown Law Technol. J. **4**(2), 523–541 (2020). https://georgetownlawtechreview.org/wp-content/uploads/2020/07/4.2-p523-541-Appel-Stark.pdf
3. Bernhard, M., et al.: Can voters detect malicious manipulation of ballot marking devices? In: 41st IEEE Symposium on Security and Privacy, pp. 679–694. IEEE (2020). https://doi.org/10.1109/SP40000.2020.00118
4. Cillizza, C.: How did Georgia get it so wrong (again)? (2020). https://www.cnn.com/2020/06/10/politics/georgia-primary-vote-brian-kemp/index.html
5. DeMillo, R., Kadel, R., Marks, M.: What voters are asked to verify affects ballot verification: a quantitative analysis of voters' memories of their ballots (2018). https://doi.org/10.2139/ssrn.3292208
6. Fowler, S.: State outlines fix for error that halted election testing. Georgia Public Broadcasting (2020). https://www.gpb.org/news/2020/09/29/state-outlines-fix-for-error-halted-election-testing
7. Gilbert, J.: Ballot marking verification protocol. http://www.juangilbert.com/BallotMarkingVerificationProtocol.pdf (2019)
8. Han, Y., Jiao, J., Weissman, T.: Minimax estimation of discrete distributions. In: 2015 IEEE International Symposium on Information Theory (ISIT), pp. 2291–2295. IEEE (2015)
9. Harte, J.: Exclusive: Philadelphia's new voting machines under scrutiny in Tuesday's elections. Reuters (2020). https://in.reuters.com/article/usa-election-pennsylvania-machines/exclusive-philadelphias-new-voting-machines-under-scrutiny-in-tuesdays-elections-idINKBN2382D2
10. Haynes, A., III, M.H.: Georgia voter verification study. https://s3.documentcloud.org/documents/21017815/gvvs-report-11.pdf (2021). Accessed 31 Oct 2021
11. Hotten, R.: Volkswagen: the scandal explained. https://www.bbc.com/news/business-34324772 (2015)
12. Kortum, P., Byrne, M., Azubike, C., Roty, L.: Can voters detect errors on their printed ballots? Absolutely. https://arxiv.org/abs/2204.09780 (2022)
13. Kortum, P., Byrne, M., Whitmore, J.: Voter verification of ballot marking device ballots is a two-part question: can they? mostly, they can. do they? mostly, they don't. Election Law J. Rules Polit. Policy 243–253 (2021). https://doi.org/10.1089/elj.2020.0632
14. Los Angeles county clerk: multilingual services program (2020). https://www.lavote.net/home/voting-elections/voter-education/multilingual-services-program/multilingual-services-program
15. Mehrotra, K., Newkirk, M.: Expensive, glitchy voting machines expose 2020 hacking risks (2019). https://www.bloomberg.com/news/articles/2019-11-08/expensive-glitchy-voting-machines-expose-2020-hacking-risks
16. National federation of the blind: blind statistics (2019). https://www.nfb.org/resources/blindness-statistics
17. Perez, E., Miller, G.: Georgia state election technology acquisition: a reality check. https://trustthevote.org/wp-content/uploads/2019/03/06Mar19-OSETBriefing_GeorgiaSystemsCostAnalysis.pdf (2019)

18. Previti, E.: Northampton officials unanimously vote 'no confidence' in ExpressVote XL voting machine (2019). https://papost.org/2019/12/20/northampton-officials-unanimously-vote-no-confidence-in-expressvote-xl-voting-machine/
19. Quesenbery, W.: Why not just use pens to mark a ballot? (2018). https://civicdesign.org/why-not-just-use-pens-to-mark-a-ballot/
20. Riggall, H.: Up to 157 incorrect ballots cast on first day of early voting, cobb elections director says. Marietta Daily Journal, Ga (2022). https://news.yahoo.com/157-incorrect-ballots-cast-first-092000088.html
21. Rivest, R.: On the notion of 'software independence' in voting systems. Phil. Trans. R. Soc. A **366**(1881), 3759–3767 (2008)
22. Secretary of the commonwealth of Pennsylvania: report concerning the examination results of election systems and software EVS 6012 with DS200 precinct scanner, DS450 and DS850 central scanners, ExpressVote HW 2.1 marker and tabulator, ExpressVote XL tabulator and electionware EMS. https://www.dos.pa.gov/VotingElections/Documents/Voting%20Systems/ESS%20EVS%206021/EVS%206021%20Secretary%27s%20Report%20Signed%20-%20Including%20Attachments.pdf (2018)
23. Shortell, T., Tatu, C.: Here's why northampton county's voting machines went wrong, county executive says. The Morning Call 12 December 2019 (2019)
24. Sneed, T.: Will L.A.'s voting overhaul be an industry disrupter or the next election debacle? (2020). https://talkingpointsmemo.com/news/will-l-a-s-voting-overhaul-be-an-industry-disrupter-or-the-next-election-debacle
25. Stark, P., Wagner, D.: Evidence-based elections. IEEE Secur. Priv. **10**, 33–41 (2012)
26. U.S. Census Bureau: city and town population totals: 2010–2020 (2020). https://www2.census.gov/programs-surveys/popest/datasets/2010-2020/cities/
27. U.S. Election Assistance Commission: voluntary voting systems guidelines 1.0, December 2005. https://www.eac.gov/sites/default/files/eac_assets/1/28/VVSG.1.0_Volume_1.PDF
28. U.S. Election Assistance Commission: voluntary voting system guidlines version 1.1. U.S. Election Assistance Commission (2015). https://www.eac.gov/sites/default/files/eac_assets/1/28/VVSG.1.1.VOL.1.FINAL1.pdf
29. U.S. Election Assistance Commission: The U.S. election assistance commission's 2020 election administration and voting survey (EAVS) (2020). https://eavsportal.com/
30. U.S. Election Assistance Commission: report of investigation, dominion voting systems D-suite 5.5-B. Williamson County, Tennessee. https://www.eac.gov/sites/default/files/TestingCertification/EAC_Report_of_Investigation_Dominion_DSuite_5.5_B.pdf (2022)
31. U.S. Senate Intelligence Committee: Russian targeting of election infrastructure during the 2016 election: summary of initial findings and recommendations. https://www.burr.senate.gov/imo/media/doc/RussRptInstlmt1-%20ElecSec%20Findings,Recs2.pdf (2018). Accessed 3 June 2020
32. Wallach, D.: On the security of ballot marking devices. Ohio State Technol. Law J. **16**(2), 558–586 (2020)
33. Zetter, K.: Los Angeles County's risky voting experiment (2020). https://www.politico.com/news/2020/03/03/los-angeles-county-voting-experiment-119157

Individual Verifiability with Return Codes: Manipulation Detection Efficacy

Paul Tim Thürwächter[1], Melanie Volkamer[1](✉), and Oksana Kulyk[2]

[1] Karlsruhe Institute of Technology, Karlsruhe, Germany
paul.thuerwaechter@student.kit.edu, melanie.volkamer@kit.edu
[2] IT University of Copenhagen, Copenhagen, Denmark
okku@itu.dk

Abstract. Researchers advocate for end-to-end verifiable voting schemes to maximise election integrity. At E-Vote-ID 2021, Kulyk et al. proposed to extend the verifiable scheme used in Switzerland (called original scheme) by voting codes to improve it with respect to vote secrecy. While the authors evaluated the general usability of their proposal, they did not evaluate its efficacy with respect to manipulation detection by voters. To close this gap, we conducted a corresponding user study. Furthermore, we study the effect of a video intervention (describing the vote casting process including individual verifiabilty steps) on the manipulation detection rate. We found that 65% of those receiving the video detected the manipulation and informed the support. If we only consider those who stated they (partially) watched the video the rate is 75%. The detection rate for those not having provided the video is 63%. While these rates are significantly higher than the 10% detection rate reported in related work for the original system, we discuss how to further increase the detection rate.

Keywords: End-to-end verifiability · Usability · Individual verifiability · Deceptive study · Manipulation detection rate

1 Introduction

Cryptographic end-to-end (E2E) verifiability facilitates detection of violations of the election integrity. From a usability perspective, individual verifiability (i.e. the ability to verify that the vote is cast as intended and stored as cast) is particularly challenging, as voters need to verify themselves. This is essential to preserve the secrecy of the vote. Thus, with E2E verifiability, in theory, it is possible to detect if voters are modified at any point in time, but only when voters know how to perform the individual verifiability and actually do so.

A range of manipulation-detection efficacy studies have been carried out to evaluate whether voters would detect if their vote is manipulated, e.g. in [11,22]. The corresponding user studies are conducted with different electronic voting systems in mind as well as with different types of attacks. In the user

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 139–156, 2022.
https://doi.org/10.1007/978-3-031-15911-4_9

study, participants are told that the usability of an electronic voting system is evaluated. However, they interact with one (or a corresponding mockup) that an attacker could have set up to make voters believe their vote is not manipulated while it is actually either altered before being sent to the ballot box or not being sent to the ballot box at all. Note, the concrete strategy an attacker could take to do so, depends on the voting system under consideration. Furthermore, not all strategies are the same, but some are more difficult for voters to detect than others. Correspondingly there is a broad range of detection rates being reported in the literature, e.g. in [18], authors report for one system a detection rate of 100% for a more easy to detect manipulation and 10% for a difficult to detect one for another system.

Our focus is on those attacks that are difficult to detect as an attacker is more likely to take those. Furthermore, our focus is on the voting system used in Switzerland (which is based on polling sheets with return and confirmation codes to enable voters to verify their vote) – more precisely, the improvement proposed at E-Vote-ID 2021 by Kulyk et al. [16]. The authors proposed to use QR Codes to enter codes and to use so-called voting codes in order to improve the guarantees with respect to the secrecy of the vote. They also conducted a user study in which they observed that their proposal has no negative impact on the general usability compared to the original scheme. Our research has the following goals:

- Improving the proposal of [16] (i.e. voting material and mockups of the voting interfaces) based on the feedback reported in their paper.
- Evaluating the manipulation detection efficacy of this improved proposal and comparing the detection rate with the one from the original scheme (note, to do so, we use the data from a similar study reported on in [18]).
- Studying the impact of providing voters additional information material about the vote casting process. We decided to use a video describing how to proceed to cast and verify votes as additional information material.

We conducted a user study with 50 participants. Our improved version of the Kulyk et al. proposal from [16] performed significantly better with respect to manipulation detection (63% detection rate) compared to the original system (10% detection rate, reported in [18]). While the detection rate for those participants who actually (at least partially) watched the video increased to 75%, it did not increase significantly. We discuss our findings in light of related work and deduce research directions for future work.

As a side contribution, our results confirm the conclusions from Kulyk et al. in [16] that, QR-code based code voting should be employed in certain types of election, as it avoids reliance on trustworthy voting clients. While they only argued based on the general usability, we showed that it has a positive effect on the manipulation detection rate too.

2 Related Work

Human aspects of verifiable voting systems have been a subject of several investigation, focusing on different aspects of verifiability, such as voters' attitudes, mental models and misconceptions of verifiability [1,2,4,9,21,22] or the usability of the actual verification process [1–5,9–11,15,17,19,20,22,26,28,30,31].

In particular, several studies focused on the effectiveness of verification procedure in different e-voting systems [2,7,18,24]. These studies show mixed results, showing that in several of the investigated systems, a significant amount of voters is not able to perform the verification correctly, thus being unable to tell whether their votes are being manipulated – e.g. less than half of participants were able to verify their votes using the Helios and the Scantegrity II voting system in the study by Acemyan et al. [2]. Other systems have demonstrated more promising results. In particular, the evaluations of voting systems implementing verification procedures based on the so-called check codes have shown a high level of verification efficiency in the studies that evaluated verification effectiveness by introducing vote manipulations in the experimental procedure and testing whether the participants of the experiment are able to detect these manipulations via the corresponding verification [18,24]. The studies in both of these works have shown a 100% verification efficiency rate with different variants of such code-based systems, meaning that all participants in their experiments were able to successfully verify their vote and detect manipulations. However, when different kinds of attacks were considered – in particular, with the adversary being able to modify the user interfaces with the goal of confusing the voter and preventing them from performing or correctly interpreting the verification results – the success rates for the verification decreased again, with only up to 56% of participants being able to detect such an attack according to the study in [18]. These studies conclude that evaluating verification efficacy via empirical experiments is crucial in understanding the security of proposed e-voting systems.

Aside from evaluating verifiability from the human factors point of view, a number of studies focused on other techniques that are introduced to e-voting systems to enhance their security – namely, to the code-voting approach [6,8,13,14,27], aimed to decrease the need to trust the voting client with regards to vote secrecy. As such, the usability of such systems has been evaluated in [17,23], showing that code voting in general can be made usable and acceptable by the voters. However, only limited evaluations of the usability of verification in code-voting systems have been conducted; one such system has been the subject of the study by Kulyk et al. [16], showing high effectiveness in terms of voters being able to cast the vote using the system, however, the study only tested the system in absence of vote manipulations.

3 Background

3.1 Swiss Electronic Voting System

Our focus is on the Swiss voting system from the Swiss Post[1]. The process to cast a vote with this system[2] is as follows: Voters receive an individual code sheet (also called polling sheet) via postal service. This polling sheet contains one initialisation code, check codes for each voting option, one confirmation code, and one finalisation code. All cotes are different for each voter. As there is no electronic ID in Switzerland, the system generates an election specific election key pair for all voters – one pair for each voter. The voters' private key is indirectly provided to them in the polling sheet. The private key can be deduced from the initialisation code.

An overview is depicted in Fig. 1a. To start the vote casting process, voters open the election webpage (the URL is provided on the polling sheet). Next, they manually enter their initialisation code (i.e. by typing the corresponding characters in the corresponding field of the webpage). Afterwards, voters select their voting option using the election webpage, i.e. clicking the option they want to select. Next, the election webpages displays a check code. According to the description on the polling sheet), voters are supposed to compare this code with the one next to their voting option on their polling sheet. The result of this check can be a pass or a fail: If both codes are the same, the voter confirms his by manually entering the confirmation code.

In case, the check was passed and the confirmation code was correctly entered, voters are supposed to receive a finalisation code. According to the polling sheet, voters are supposed to check whether such a code is displayed and whether it matches the one on their polling sheet. Only if this second check is passed, voters can be assured that their vote has been stored as intended (i.e. cast as intended plus stored as cast). The voting scheme provides individual verifiability under the assumption that the printing server and the voting client do not collaborate. Note, we are aware that the implementation when it comes to the universal verifiability had severe shortcomings, see e.g. [12]. We believe that these are issues the company faced when implementing the underlying cryptographic primitives. Thus, it can be fixed and as such it is still worth to study the individual verifiabilty of schemes like the one used in Switzerland.

We refer to this system incl. the election material and user interfaces as **'original system'**.

3.2 E-Vote-ID-2021-Proposal

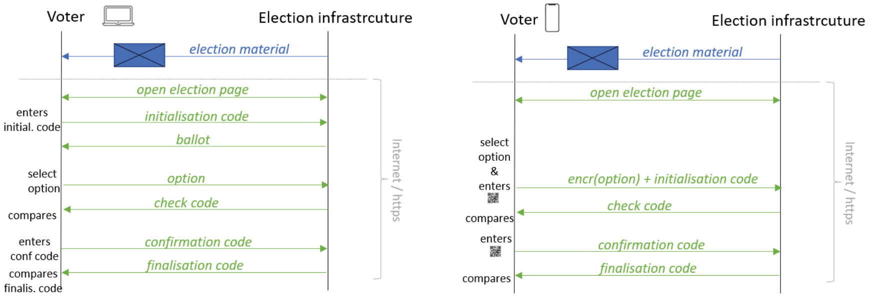
At E-Vote-ID-2021, Kulyk et al. proposed in [16] to extend the Swiss voting system by individual voting codes. With this extension, the individual polling

[1] https://evoting-community.post.ch/de?_ga=2.79449501.804715002.1658647288-420296842.1658647288.

[2] Note, the system is used for polls in which voters select *1 out of n* options. Usually 2–3 of such polls are conducted at the same time. For our research, we assume that there is only one poll.

sheet from the Swiss system also contains one voting code per option. The voting codes are different for each voter. Thus, voters are supposed to enter the voting code representing their chosen option (instead of clicking on the option they want to selection on the election webpage). As the voting client cannot map the voting code to any of the options, the assumption on the trustworthy voting client is no longer needed[3].

To address shortcoming of entering long voting codes, the authors proposed that voters use their camera-equipped smartphones, to cast a vote by scanning a corresponding QR code (containing the voting code). Their proposed (simplified) scheme is depicted in Fig. 1b. To integrate these ideas, the authors adopted the voting material and the election webpage from [18] accordingly. In particular, they introduce voting cards which contained on the front page the voting code as QR-code and on the back page the option and the corresponding return code.



(a) Swiss voting scheme

(b) E-Vote-ID-2021-proposal

Fig. 1. Vote casting

In this paper, we refer to the proposal incl. the election material and user interfaces from Kulyk et al. as **'E-Vote-ID-2021-proposal'**.

4 Improvements to E-Vote-ID-2021-Proposal and Descriptive Video

4.1 Improvements to the Voting Material and User Interfaces

Based on the feedback Kulyk et al. received from their participants, we deduced the following improvements:

- Providing the URL to the election webpage not only as text but also as a QR-code to make it easier for voters to open the correct election webpage.

[3] Note, however, that one needs to ensure that the mapping of the voting codes to options for each voter remains secret to the adversary. Therefore it is important that the printers are operated offline as they need to be fully trustworthy.

- In order to scan the voting code, users had to scan two QR-codes at once: the one on the polling sheet and the one the voting card (which had to be placed above each other). Participants were confused as they were not aware that one can scan two QR-codes at the same time. Kulyk et al. proposed to have the QR-code on the polling sheet to make sure participants scan the voting code only when placed there[4]. We decided to trust voters to put it there. Furthermore, we added a tick-box on the start page of the vote casting interfaces where voters would need to confirm that they properly placed it. Thus, we could remove the second QR-code to make it less confusing for voters.
- Participants were missing that they have to confirm that they cast their vote on their own and were not observed (as this is the case with postal voting). We added such a confirmation statement.
- The user interfaces of Kulyk et al. did contain minimal information. Their motivation to do so was that voters should anyway follow the instructions on the polling sheet. However, participants were complaining that the interface looked not very trustworthy due to the minimal amount of text. Therefore, we added the instructions from the polling sheet also on the user interfaces of the election webpage. We are aware that this only increases perceived security but without this adoption we would ignore users' feedback. Furthermore perceived security is likely to influence voters' trust in the voting system in place. At the end, we need to achieve both: Having a trustworthy end-to-end verifiable voting system in place which voters trust.

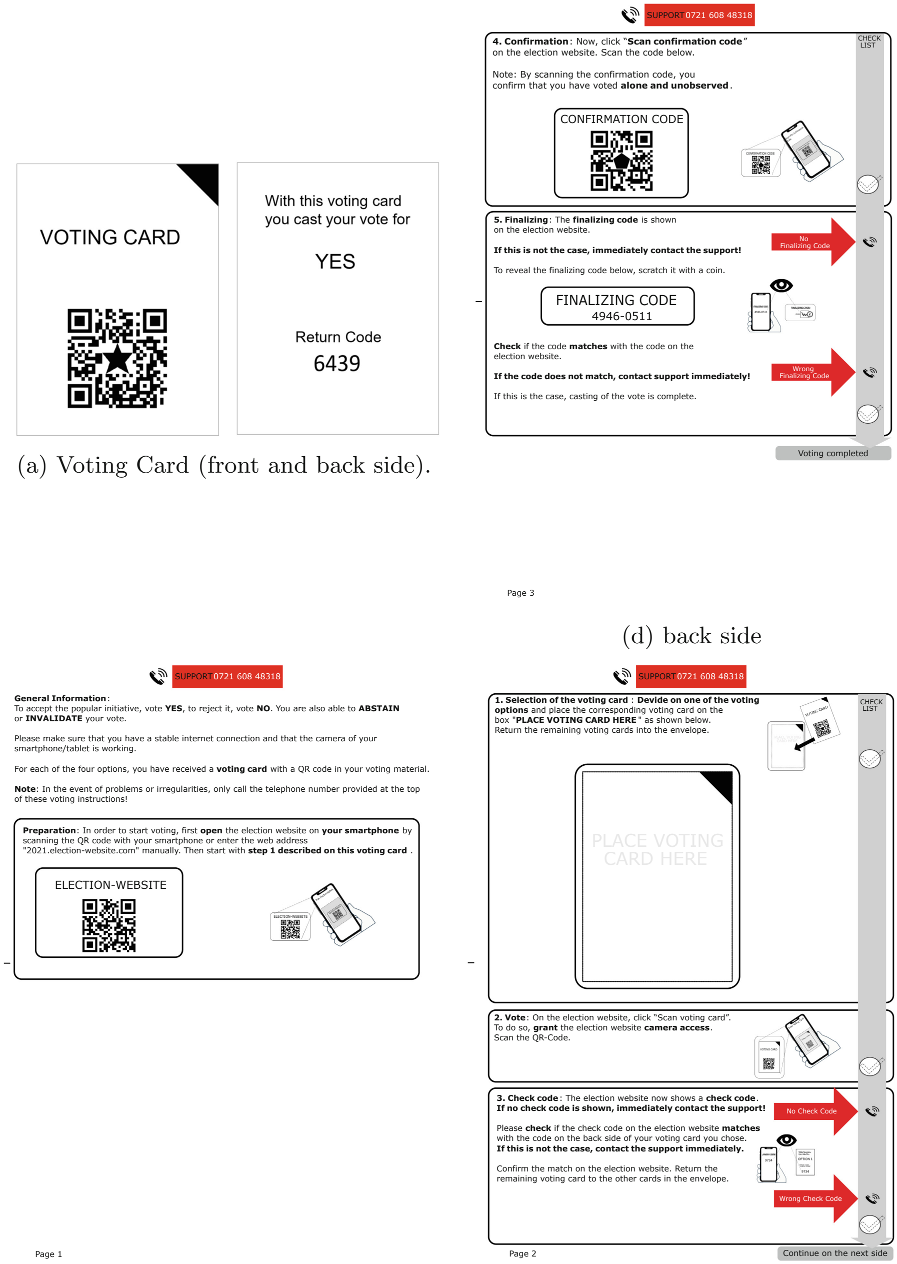
The modified voting material and user interfaces are depicted in Fig. 2 and 3. In this paper, we refer to this system as improved-proposal.

4.2 Descriptive Video

We were discussing what additional information voters may receive or have access to, regarding the online voting channel. There might be discussion forums, information about the company who provides the systems, the setup, maybe also about security evaluations in case there are some. In addition, we expect that their are videos describing the vote casting process to give voters an idea of the process and maybe what to particular care about. As we thought the first list might be very much related to the actual system in place, it is worth studying the impact of a video describing the process. Note, we decided to go with a video which provides the necessary information in a one-two lines text field rather than with audio, as we did not want that the audio is an issue when conducting the study (see Sect. 5).

The video therefore shows all the steps from receiving and opening the voting material to checking the finalisation code. The video takes 9 min. It also highlights twice the number of the support to be called. The reason it takes 9 min is that it gives the recipients time to read the text in the polling sheet at the

[4] Fore the exact reasons, we refer the reader to [16].



VOTING CARD

With this voting card you cast your vote for

YES

Return Code

6439

(a) Voting Card (front and back side).

☎ SUPPORT 0721 608 48318

CHECK LIST

4. Confirmation: Now, click "**Scan confirmation code**" on the election website. Scan the code below.

Note: By scanning the confirmation code, you confirm that you have voted **alone and unobserved**.

CONFIRMATION CODE

5. Finalizing: The **finalizing code** is shown on the election website.

If this is not the case, immediately contact the support!

No Finalizing Code

To reveal the finalizing code below, scratch it with a coin.

FINALIZING CODE

4946-0511

Check if the code **matches** with the code on the election website.

If the code does not match, contact support immediately!

Wrong Finalizing Code

If this is the case, casting of the vote is complete.

Voting completed

Page 3

(d) back side

☎ SUPPORT 0721 608 48318

General Information:
To accept the popular initiative, vote **YES**, to reject it, vote **NO**. You are also able to **ABSTAIN** or **INVALIDATE** your vote.

Please make sure that you have a stable internet connection and that the camera of your smartphone/tablet is working.

For each of the four options, you have received a **voting card** with a QR code in your voting material.

Note: In the event of problems or irregularities, only call the telephone number provided at the top of these voting instructions!

Preparation: In order to start voting, first **open** the election website on **your smartphone** by scanning the QR code with your smartphone or enter the web address "2021.election-website.com" manually. Then start with **step 1 described on this voting card** .

ELECTION-WEBSITE

Page 1

☎ SUPPORT 0721 608 48318

CHECK LIST

1. Selection of the voting card : **Devide on one of the voting options** and place the corresponding voting card on the box "**PLACE VOTING CARD HERE**" as shown below. Return the remaining voting cards into the envelope.

PLACE VOTING CARD HERE

2. Vote: On the election website, click "Scan voting card". To do so, **grant** the election website **camera access**. Scan the QR-Code.

3. Check code: The election website now shows a **check code**.
If no check code is shown, immediately contact the support!

No Check Code

Please **check** if the check code on the election website **matches** with the code on the back side of your voting card you chose.
If this is not the case, contact the support immediately.

Confirm the match on the election website. Return the remaining voting card to the other cards in the envelope.

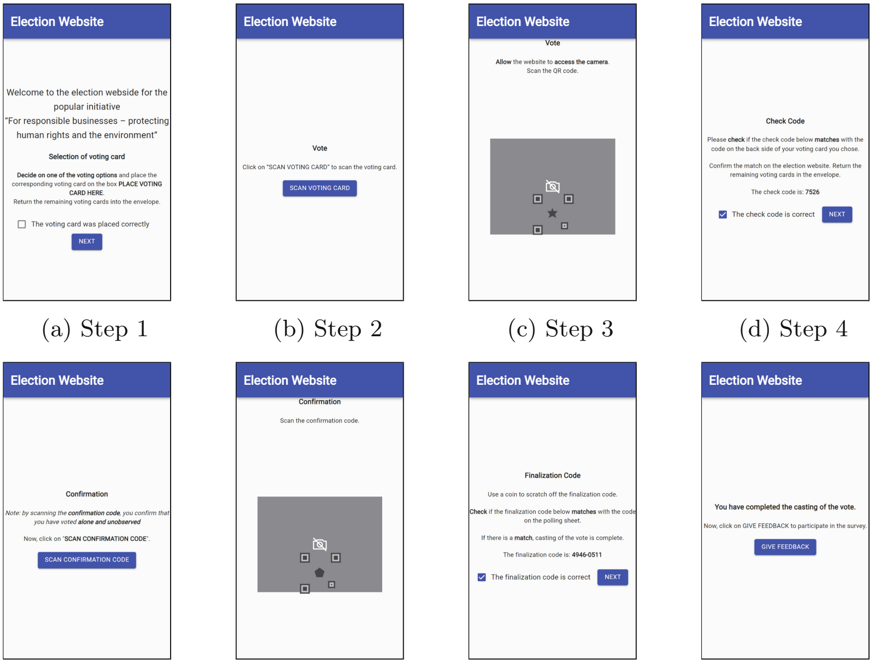
Wrong Check Code

Page 2

Continue on the next side

(b) inner - left (c) inner - right

Fig. 2. Polling sheet (b–d) with the scratch field being removed in (d); and voting cards (a)



Welcome to the election webside for the popular initiative
"For responsible businesses – protecting human rights and the environment"

Selection of voting card

Decide on one of the voting options and place the corresponding voting card on the box **PLACE VOTING CARD HERE**.
Return the remaining voting cards into the envelope.

□ The voting card was placed correctly

(a) Step 1

Vote

Click on "SCAN VOTING CARD" to scan the voting card.

(b) Step 2

Vote

Allow the website to **access the camera**.
Scan the QR code.

(c) Step 3

Check Code

Please **check** if the check code below **matches** with the code on the back side of your voting card you chose.

Confirm the match on the election website. Return the remaining voting cards in the envelope.

The check code is: **7526**

☑ The check code is correct

(d) Step 4

Confirmation

*Note: by scanning the **confirmation code**, you confirm that you have voted **alone and unobserved***

Now, click on "**SCAN CONFIRMATION CODE**".

(e) Step 5

Confirmation

Scan the confirmation code.

(f) Step 6

Finalization Code

Use a coin to scratch off the finalization code.

Check if the finalization code below **matches** with the code on the polling sheet.

If there is a **match**, casting of the vote is complete.

The finalization code is: **4946-0511**

☑ The finalization code is correct

(g) Step 7

You have completed the casting of the vote.

Now, click on GIVE FEEDBACK to participate in the survey.

(h) Step 8

Fig. 3. Voting webpage. Note, the steps, we refer to, correspond to those in the polling sheet.

same time, i.e. reading one paragraph or one step before continuing, i.e. actually conducting this step in the video. The video is available online[5].

5 Methodology

We first introduce our research questions and corresponding hypotheses, afterwards we describe the study procedure before discussing ethics, how we meet data protection regulations, and how we recruited our participants.

5.1 Research Questions, Hypotheses

The proposal from [16] improves the security level of the original scheme. The general usability was shown to be similar to the original system. An open question remains, however, how this idea perform with respect to the manipulation-detection efficacy - both with and without providing a descriptive video. Correspondingly, we define the following research questions:

[5] https://youtu.be/Yj7yz437OEc.

How does the improved-proposal performs in terms of manipulation-detection efficacy (measured as the rate of participants detecting the manipulation of their vote) with and without watching the video?

The authors of [16] based their voting material and election webpage on the improvements from [18]. We further improved both based on the feedback the authors reported on in [16]. The improvements of [18] resulted in a significantly higher manipulation detection rate than original system. Therefore, we expect that our improvements of the improved-proposal outperform the original system with respect to manipulation-detection efficacy. We therefore define the following hypotheses:

H_1: The improved-proposal without interventions has a significantly higher manipulation-detection efficacy than the original system.

H_2: The improved-proposal in combination with the watching the video has a significantly higher manipulation-detection efficacy than the original system.

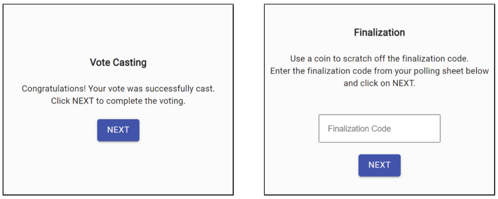
Note, the validation of this hypotheses come with some limitations as we collected only data for the improved-proposal (with and without the video) while we use for the original system the date from [18]. We discuss this further in the limitation section.

In particular for people using the improved-proposal the first time, the video helps to give them a better idea about the vote casting including scanning and verifying the various codes. The video in particular indicates that the support should be contacted in case the shown codes do not match the expected ones. We therefore define the following hypothesis:

H_3: The improved-proposal in combination with the watching the video has a significantly higher manipulation-detection efficacy than the improved-proposal without further descriptions or explanations.

5.2 Considered Manipulation-Types

Kulyk et al. studied two different types of manipulations in [18]. One of their attacks would not be possible in the proposal from Kulyk et al. [16]: Adversaries would need to know the voting code for the option they want to cast a vote for – which is not the case by design of any code voting scheme. In the other one, adversaries attempt to nullify cast votes by not sending the voting code to the election infrastructure and manipulating the voting client with the purpose to convince the voter that their vote has been cast successfully. For this attack after entering the voting-code, the election webpage would confirm the correctness of the check code. Furthermore, it would state that the check code is correct and that one can continue to finish the vote casting process. Note, it is not possible for the adversary to show the finalisation code as they cannot send a valid voting code to the election infrastructure. Therefore, the adversary would need to change these steps, too: Instead of asking voters to compare the displayed finalisation code with the one in the polling sheet, the manipulated voting client could ask voters to enter the finalisation code. Figure 4 shows the content of the manipulated interfaces.



Vote Casting

Congratulations! Your vote was successfully cast.
Click NEXT to complete the voting.

Finalization

Use a coin to scratch off the finalization code.
Enter the finalization code from your polling sheet below and click on NEXT.

(a) Step 4

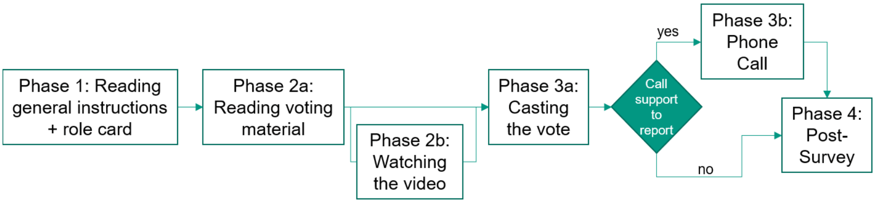
(b) Step 7

Fig. 4. Manipulated webpage, only displaying steps that are different, any only the actual text/UI elements.

5.3 Study Procedure

Figure 5 depicts an overview of the study procedure. The study was conducted in German. Voting material and election webpage were translated for this paper. Furthermore, it was a remote study. The ballot of the election we simulated for our study contained four options. Participants were randomly assigned to one of the two groups: The no-video-group and the video-group. Participants received the study material in an envelope either via postal service or from someone they know. The following content was included:

- A study letter describing the study, the time frame, which other material is included in the envelope, the conditions incl. the next steps to take, and information that they can cancel participation at any time. Note, in a footnote, the link to the post-survey was included.
- Role card explaining who they should suppose to be for the study and which option to vote for.
- Envelope with the actual voting material, i.e.,
  - the election letter from the election officials which recommended to first read the polling sheet before starting the vote casting process. Furthermore, it mentions that in case of problems or questions they should call the (study) support. For participants in the video-group this document also recommended to first watch a descriptive video. A corresponding link was provided.
  - the polling sheet; and
  - the voting cards with the voting-code.



Phase 1: Reading general instructions + role card
Phase 2a: Reading voting material
Phase 2b: Watching the video
Phase 3a: Casting the vote
Call support to report
yes
no
Phase 3b: Phone Call
Phase 4: Post-Survey

Fig. 5. Study procedure for both groups (with and without the link to the video)

Participants were supposed to open the envelope, and read the study letter and the role card (phase 1). Afterwards, they were supposed to open the inner envelope with the voting material and to read the polling sheet (phase 2a). Participants in the video-group were asked to then watch the video (phase 2b). Afterwards, participants were supposed to start casting their vote (phase 3a). Both groups (the one with and without the link to the video) got the manipulated interfaces as described in Sect. 5.2.

In the case that participants did not notice the manipulation or have noticed it but did not call the support, they could just finish vote casting. After having finished the vote casting process, the election webpage displayed the link to the post-survey (phase 4). This survey, first, provides information about the study and data collection. It contained the informed consent. Once consent was provided, participants were debriefed. If they decide to continue with the survey, they were asked whether they detected the manipulation they read about in the debriefing text. Note, the question on the manipulation detection had three options: (1) I noticed it and I called the (study) support, (2) I noticed it but I did not call the (study) support, and (3) I did not notice the manipulation. In case the first option was selected, this had to be confirmed by entering the number 22. Those who called the (study) support got this number on the phone once they have reported the manipulation they observed. In case the second option was selected, participants were asked an additional open text question on why they did not call the support. The survey also asked whether they first read the instructions on the polling sheet before starting the vote casting process and we asked demographic questions. Participants in the video-group got additional questions: Whether they watched the video (entirely) and whether it was helpful (to detect the manipulation).

In the case that study participants did notice the manipulation and called the (study) support[6] (phase 3b), the support first asked them to provide details about the issues they have. The goal was to first make sure they actually observed the manipulation and to find out to which group they were assigned. The study support took a note of this. Afterwards the participant who called was debriefed on the phone. If they decided to continue with the study, they were provided with the link to the post-survey[7] and with the number 22 needed for the post-survey. Finally, the study support thanked the caller.

5.4 Ethics, Data Protection, Recruitment

The study was announced with the goal to evaluate the usability of an online voting system. Thus, one may call it a deceptive study. Therefore, the study was approved by the ethic committee of our university. Their checks contain legal issues as well. As such the compliance with data protection laws was attested too. We still want to comment on some important aspects: The postal addresses were

[6] Note, in case the study support could not answer the call, this person was called back as soon as possible. All telephone numbers were deleted afterwards.

[7] The post-survey was the same for those not detecting all participants.

deleted once they were put on the envelopes. For the survey we used SocSciSurvey which is GDPR compliant[8]. Participants were debriefed either on the phone or through the post-survey. The study material contained a telephone number and an email address to get in touch with us in case of general questions regarding the study or any doubts.

Participants were recruited in various different ways: Public channels, social media, friends of friends (in case they were not aware of our research) as well as through a snow-ball principle, asking those who agreed to participate to announce it to friends and family, too. Due to the remote study setting, we decided to not offer a reimbursement.

6 Results

We sent out the voting instructions to 60 people (30 for each group). Eventually, a total of 50 people completed the post-survey (24 assigned to the no-video-group and 26 to the video-group). Table 1 shows their demographics, as well as of the participants in the study from [18] for the sake of comparison[9]. All statistical calculations for our hypotheses are performed using R packages "stats" and "rstatix". We report our results without corrections for multiple comparisons.

Table 1. Demographics of participants for age Mean/SD and gender

Experiment	Age	Gender
From [18]	34.34/15.54	66F, 62M
Our study	27.5/10.135	26F, 24M

6.1 Overall Manipulation Detection

Overall, 32 out of 50 participants reported detecting the manipulation. Of them, 17 were in the video-group and 15 were in the no-video-group, leading to detection rates of 65.4% and 62.5% correspondingly. We used Fischer's exact test [25] for the evaluation of our hypotheses, as commonly recommended for categorical data with 2 × 2 contingency tables with small sample sizes. Both of the groups had significantly higher detection rates compared to the original system (which according to [18] had a detection rate of 10%), as shown by Fisher's exact test[10] **confirming** H_1 ($OR = 13.98$, 95% CI $= [3.02, Inf]$, $p = .0004$) **and** H_2 ($OR = 15.83$, 95% CI $= [3.466, Inf]$, $p = .0002$). No significant differences were detected between the no-video-group and the video-group (Fisher's test, $OR = 1.13$, 95% CI $= [0.3679725, Inf]$, $p = .532$), thus **failing to confirm** H_3.

[8] Data protection policy: https://www.soscisurvey.de/en/data-protection .

[9] Note, the authors of [18] do not report the demographics separate for their groups.

[10] Note that for all our hypotheses one-tailed tests are used.

6.2 Manipulation Detection for Various Subgroups

The free-text answers were analysed by two of the authors independently and then discussed. As the provided free-text answers were rather short we took this approach rather than a formal open-coding approach. Eight participants (4 in the no-video-group and 4 in the video-group) answered in the post-survey that they noticed the manipulation, but did not report it to the study examiner. When asked to explain why they did not report it, the following reasons were stated[11]: Two mentioned that they did not want to call in the late hour, two answered that calling would be too much effort, two believed that the vote casting was successfully completed despite the fact that the steps on the interface did not match the ones on the polling sheet, two thought that they themselves were at fault and one believed that the missing code is displayed later in the process.

Overall 31 participants (17 in the video-group and 14 in the no-video-group) reported reading the voting material before starting with the voting procedure; 21 (ten in the video-group and 11 in the no-video-group) reported reading the materials while voting. Note that two participants reported both reading the materials completely beforehand and reading them again while voting. One participant (from the video-group) reported reading the study instructions and role card beforehand, but only reading the polling sheet while voting. Of the 31 participants who read the voting materials beforehand, 23 (74.1%) detected and reported the manipulation, as opposed to 9 out of 19 (47.4%) of those who did not read the materials beforehand.

6.3 Video Related Statements

Out of 26 participants assigned to the video-group, nine reported watching parts of the video, 11 reported watching all of it and six reported not watching the video at all. None of the participants reported watching the video more than once. The participants who reported not watching the video gave the following reasons for this: Two answered that the video was too long, one answered that watching the video would be too much effort and three answered that they believed watching the video was not necessary to complete the voting.

From those 20 participants who stated that they fully or partially watch the video, 15 participants reported the manipulation and called the support. In particular, nine out of 11 of participants that watched the video entirely reported the manipulation, compared to six out of nine of participants who watched parts of the video.

Furthermore, one could observe differences between manipulation detection rate depending on whether the participants familiarised themselves with the voting procedure before starting voting, either by watching the video fully (in the video-group) or by reading the voting materials beforehand (in both video-group and no-video-group). As such, 24 out of 34 participants who either watched the video fully or read the voting materials before voting were able to detect the

[11] Note that some of the participants mentioned several reasons for not calling.

manipulation (70.6%) as opposed to 8 out of 16 (50%) participants who did neither of these things.

Out of the participants who reported watching the video either fully or partially, who also have detected the manipulation and called the support (15 participants), the following answers were given regarding to whether the video was helpful to them for detecting the manipulation: nine agreed, four disagreed, and one were neutral.

7 Discussion

Our results clearly show that the E-Vote-ID-2021-proposal outperforms the original system with respect to the detection manipulation rate (62.5% to 10%). As the authors in [16] showed that the E-Vote-ID-2021-proposal outperforms the original system with respect to the provided guarantees for vote secrecy (because it uses voting codes) and that they have a similar general usability performance, it can clearly be recommended to consider the E-Vote-ID-2021-proposal for the elections and polls in Switzerland as well as for any other election contexts with simple ballots. Note, the proposal of Kulyk et al. [16] does also outperform the original one because (1) the assumption that the vote casting device is not violating vote secrecy is not needed and (2) it is only possible to conduct limited election integrity related attacks as one can only remove votes but not change them – thus large scale manipulations would result in a unexpected low turnout.

The findings from the free-text answers (e.g. they thought it is too much effort to call, they thought they made a mistake) indicate that increasing the manipulation rate would need additional measures such as awareness raising for verifiability and why the voting material received via postal service can be trusted but not necessary the election webpage. This is a clear and important direction for future work.

We also found that participants who reported that they read the voting material only as they voted were more likely to follow the instructions on their screen, thus missing the manipulation. Thus, those who familiarized themselves with the process beforehand are more likely to detect the manipulation (between 75% and 77% compared to 62.5% and 65%). Thus, in particular in contexts like in Switzerland in which elections and/or polls happen several times a year, it gets over time more likely that manipulations are detected: If we assume that voters have voted several times with a system that is not manipulated and thus get familiar with the correct process, they might be more likely to detect a manipulation with future elections than if already the first time the system is in place, it got manipulated. The evaluation of such hypotheses is part of our future work.

Our study furthermore detected higher rates of manipulation detection compared to related work evaluating same kind of attacks - as such, the study by Kulyk et al. [18] found 43% manipulation detection rates and the study by Volkamer et al. [29] reported 41% compared to 62.5% of participants in our study (those who did not watch the video). One explanation could be the difference in demographics, in particular, the fact that the participants in our study tended to be younger than in related work, see e.g. Table 1 for the comparison between our participants and those in [18]. A study on the effects of demographic factors, including but not limited to age, gender and education, on the voter's ability to detect manipulations is therefore an interesting direction of future work. An other one might be that less people in [18,29] have read the instructions before starting the vote casting process. Note, a comparison is not possible as the authors do not provide any related information.

Study Limitations: Our study has similar limitations to other user studies evaluating the manipulation-detection efficacy in verifiable electronic voting: It is about a mock election and no actual election. Participants cast a vote for the option they were asked to select. Thus, this vote is not very personal to them or important. Participating in a study and, thus, agreeing to take time for it may result in spending more time in reading the instructions compared to casting a vote in an actual election. However, introducing vote manipulations in an actual election to measure manipulation-detection efficacy would pose critical ethical and legal issues. Thus, there is not much one can do about it.

Another limitations of all these studies evaluating manipulation-detection efficacy (including ours) is that we need to trust that those few participants who know each other have not informed others about the manipulation. Furthermore, we evaluated the scheme in Germany with participants who have not cast a vote with the original system. The results may be different for participants who are familiar with the original system.

We studied one implementation of adversaries' attempt to make voters believe their vote was cast as intended while their vote is not considered in the tally. The details can vary, i.e., the text displayed to convince voters that everything is fine. As future work, one could study the attack with different text.

In order to test two of the three hypotheses, we used data from our previous paper, i.e. [18]. This comes with some limitations as the study in the previous paper was a lab study, i.e. the study instructor was in the same room while in this paper, the study instructor could only be reached via phone. However, on the one hand the difference with respect to the detection rates are large and several studies have already shown the issues with the original system. Therefore, we wanted to focus our own data collection on the new proposal and the effect of the video.

8 Conclusion

Verifiable voting schemes are the de-facto standard when considering online voting for political elections. At the same time, the verifiable voting systems in place only provide vote secrecy if the voting client is trustworthy. While this shortcoming can be addressed with code voting, such approaches are currently not considered, as the community and election officials are concerned about the usability implications. Kulyk et al. demonstrated in [16] that a code voting based extension of the original system can be as usable as the original one. We underline their conclusions as we show that such an extension can also significantly increase the manipulation-detection efficacy. Thus, it is worth considering code-voting verifiable voting schemes, as the cumbersome steps of entering voting codes manually can be replaced by easy-enough steps – i.e., scanning QR codes – without significantly reducing the usability while enabling systems with higher security guarantees. Thus, our research should encourage more research on combining code-voting with verifiable schemes.

While the manipulation-detection efficacy is significant higher for the studied scheme compared to the original system one, there is room for improvements. We evaluated whether a video intervention describing the vote casting steps including those to verify can further improve this rate. While we observed some increase, it was not significant. Based on the discussion of our results, we conclude that it is important to study various types of interventions with respect to their effect on manipulation-detection efficacy. In particular, approaches explaining the importance of verifiability should be developed and evaluated.

Acknowledgements. This research was further supported by funding from the topic Engineering Secure Systems, subtopic 46.23.01 Methods for Engineering Secure Systems, of the Helmholtz Association (HGF) and by KASTEL Security Research Labs.

References

1. Acemyan, C.Z., Kortum, P., Byrne, M.D., Wallach, D.S.: Usability of voter verifiable, end-to-end voting systems: baseline data for Helios, Prêt à Voter, and Scantegrity II. USENIX J. Election Technol. Syst. **2**(3), 26–56 (2014)
2. Acemyan, C.Z., Kortum, P., Byrne, M.D., Wallach, D.S.: From error to error: why voters could not cast a ballot and verify their vote with Helios, Prêt à Voter, and Scantegrity II. USENIX J. Election Technol. Syst. **3**(2), 1–19 (2015)
3. Acemyan, C.Z., Kortum, P., Byrne, M.D., Wallach, D.S.: Summative usability assessments of STAR-Vote: a cryptographically secure e2e voting system that has been empirically proven to be easy to use. Hum. Factors **64**, 1–24 (2018)
4. Bär, M., Henrich, C., Müller-Quade, J., Röhrich, S., Stüber, C.: Real world experiences with bingo voting and a comparison of usability. In: EVT/WOTE (2008)
5. Bernhard, M., et al.: Can voters detect malicious manipulation of ballot marking devices? In: 2020 IEEE Symposium on Security and Privacy (SP), pp. 679–694. IEEE (2020)
6. Budurushi, J., Neumann, S., Olembo, M.M., Volkamer, M.: Pretty understandable democracy - a secure and understandable internet voting scheme. In: ARES, pp. 198–207 (2013)
7. Budurushi, J., Renaud, K., Volkamer, M., Woide, M.: An investigation into the usability of electronic voting systems for complex elections. Ann. Telecommun. **71**(7–8), 309–322 (2016)
8. Chaum, D.: SureVote: technical overview. In: Proceedings of the Workshop on Trustworthy Elections (WOTE 2001) (2001)
9. Distler, V., Zollinger, M.L., Lallemand, C., Roenne, P., Ryan, P., Koenig, V.: Security-visible, yet unseen? How displaying security mechanisms impacts user experience and perceived security. In: ACM CHI, pp. 605:1–605:13 (2019)

10. Fuglerud, K.S., Røssvoll, T.H.: An evaluation of web-based voting usability and accessibility. Univ. Access Inf. Soc. **11**(4), 359–373 (2012)
11. Gjøsteen, K., Lund, A.S.: An experiment on the security of the Norwegian electronic voting protocol. Ann. Telecommun. **71**(7–8), 299–307 (2016)
12. Haines, T., Lewis, S.J., Pereira, O., Teague, V.: How not to prove your election outcome. In: 2020 IEEE Symposium on Security and Privacy (SP), pp. 644–660. IEEE (2020)
13. Helbach, J., Schwenk, J.: Secure internet voting with code sheets. In: Alkassar, A., Volkamer, M. (eds.) Vote-ID 2007. LNCS, vol. 4896, pp. 166–177. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-77493-8_15
14. Joaquim, R., Ribeiro, C., Ferreira, P.: VeryVote: a voter verifiable code voting system. In: Ryan, P.Y.A., Schoenmakers, B. (eds.) Vote-ID 2009. LNCS, vol. 5767, pp. 106–121. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04135-8_7
15. Karayumak, F., Olembo, M.M., Kauer, M., Volkamer, M.: Usability analysis of Helios-an open source verifiable remote electronic voting system. In: EVT/WOTE. USENIX (2011)
16. Kulyk, O., Ludwig, J., Volkamer, M., Koenig, R.E., Locher, P.: Usable verifiable secrecy-preserving e-voting. In: Electronic Voting: 6th International Joint Conference, E-Vote-ID. University of Tartu Press (2021)
17. Kulyk, O., Neumann, S., Budurushi, J., Volkamer, M.: Nothing comes for free: how much usability can you sacrifice for security? IEEE Secur. Priv. **15**(3), 24–29 (2017)
18. Kulyk, O., Volkamer, M., Müller, M., Renaud, K.: Towards improving the efficacy of code-based verification in internet voting. In: Bernhard, M., et al. (eds.) FC 2020. LNCS, vol. 12063, pp. 291–309. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-54455-3_21
19. MacNamara, D., Gibson, P., Oakley, K.: A preliminary study on a DualVote and Prêt à voter hybrid system. In: CeDEM, p. 77 (2012)
20. MacNamara, D., Scully, T., Gibson, P.: DualVote addressing usability and verifiability issues in electronic voting systems (2011). http://www-public.it-sudparis.eu/~gibson/Research/Publications/E-Copies/MacNamaraSGCOQ11.pdf. Accessed 12 May 2022
21. Zollinger, M.-L., Estaji, E., Ryan, P.Y.A., Marky, K.: "Just for the Sake of Transparency": exploring voter mental models of verifiability. In: Krimmer, R., et al. (eds.) E-Vote-ID 2021. LNCS, vol. 12900, pp. 155–170. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86942-7_11
22. Marky, K., Kulyk, O., Renaud, K., Volkamer, M.: What did I really vote for? In: ACM CHI, p. 176 (2018)
23. Marky, K., Schmitz, M., Lange, F., Mühlhäuser, M.: Usability of code voting modalities. In: ACM CHI (2019)
24. Marky, K., Zollinger, M.L., Roenne, P., Ryan, P.Y., Grube, T., Kunze, K.: Investigating usability and user experience of individually verifiable internet voting schemes. ACM Trans. Comput.-Hum. Interact **28**(5), 1–36 (2021)
25. McDonald, J.H.: Handbook of Biological Statistics, vol. 2. Sparky House Publishing, Baltimore (2009)
26. Oostveen, A.M., Van den Besselaar, P.: Users' experiences with e-voting: a comparative case study. J. Electron. Governance **2**(4), 357–377 (2009)

27. Ryan, P.Y.A., Teague, V.: Pretty good democracy. In: Christianson, B., Malcolm, J.A., Matyáš, V., Roe, M. (eds.) Security Protocols 2009. LNCS, vol. 7028, pp. 111–130. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36213-2_15
28. Sherman, A.T., et al.: An examination of vote verification technologies: findings and experiences from the Maryland study (2006)
29. Volkamer, M., Kulyk, O., Ludwig, J., Fuhrberg, N.: Increasing security without decreasing usability: comparison of various verifiable voting systems. In: Eighteenth Symposium on Usable Privacy and Security (SOUPS 2022). USENIX Association, Boston, August 2022. https://www.usenix.org/conference/soups2022/presentation/volkamer
30. Weber, J.L., Hengartner, U.: Usability study of the open audit voting system Helios (2009). https://www.jannaweber.com/wp-content/uploads/2009/09/858Helios.pdf. 12 May 2022
31. Winckler, M., et al.: Assessing the usability of open verifiable E-voting systems: a trial with the system Prêt à voter. In: ICE-GOV, pp. 281–296 (2009)

Logic and Accuracy Testing: A Fifty-State Review

Josiah Walker(✉), Nakul Bajaj, Braden L. Crimmins, and J. Alex Halderman

University of Michigan, Ann Arbor, USA
{jhwalker,nbajaj,bradenlc,jhalderm}@umich.edu

Abstract. Pre-election logic and accuracy (L&A) testing is a process in which election officials validate the behavior of voting equipment by casting a known set of test ballots and confirming the expected results. Ideally, such testing can serve to detect certain forms of human error or fraud and help bolster voter confidence. We present the first detailed analysis of L&A testing practices across the United States. We find that while all states require L&A testing before every election, their implementations vary dramatically in scope, transparency, and rigorousness. We summarize each state's requirements and score them according to uniform criteria. We also highlight best practices and flag opportunities for improvement, in hopes of encouraging broader adoption of more effective L&A processes.

1 Introduction

The vast majority of votes in the United States are counted mechanically, either by optical scanners that read paper ballots or by direct-recording electronic (DRE) voting machines [70]. To validate that these tabulation devices are configured and functioning correctly, jurisdictions perform a procedure called "logic and accuracy testing" ("L&A testing") shortly before each election. It typically involves casting a "test deck"—a set of ballots with known votes—on each machine, then printing the results and ensuring the tally is as expected. Any deviation is a potential indicator that the election equipment has misbehaved.

While more sophisticated mechanisms such as risk-limiting audits [37], and end-to-end verification [15] can reliably detect and recover from both errors and attacks after the fact, they are not yet widely applied in the U.S . Even if they were, L&A testing would remain useful for heading off some sources of error before they affected results. Ideally, L&A testing can protect against certain kinds of malfunction, configuration error, and fraud as well as strengthen voter confidence, but its effectiveness depends on many details of how the testing is performed. In the U.S., L&A testing requirements—like most aspects of election procedure and the selection of voting equipment—are determined by individual states, resulting in a diversity of practices with widely varying utility.

Unfortunately, this heterogeneity means that many states diverge negatively from the norm and makes it difficult to offer the national public any blanket

R. Krimmer et al. (Eds.): E-Vote-ID 2022, LNCS 13553, pp. 157–184, 2022.
https://doi.org/10.1007/978-3-031-15911-4_10

assurances about the degree of protection that L&A testing affords. Moreover, many states do not publish detailed L&A procedures, leaving voters with little ability to assess the effectiveness of their own states' rules, let alone whether any tests they observe comply with them. Yet this decentralized regulatory environment has also allowed a variety of positive L&A testing procedures to evolve, and there are abundant opportunities for the exchange of best practices.

This paper provides the first comparative analysis of L&A testing requirements across the fifty states. To determine how each state performs L&A testing, we conducted an extensive review of available documentation and reached out to election officials in every state. We then assessed and scored each state's policy using criteria designed to reflect its functional effectiveness and suitability as a basis for voter confidence. The results provide a detailed understanding of how states' procedures differ and how well they approach an ideal model of what L&A testing can achieve. Our analysis reveals that several important L&A criteria are absent in many or most states' rules, yet we also highlight specific examples of policies that could serve as models for broader dissemination. We hope this work will encourage the adoption of more effective L&A testing requirements across the United States and help promote policies that better inspire public trust.

2 Background

2.1 L&A Testing Goals

L&A testing was first introduced in the early 1900s s for lever-style voting machines [63], which contained a mechanical counter for each candidate. The counters were susceptible to becoming jammed due to physical failure or tampering, so tests were designed to establish that each counter would advance when voted.

Modern DRE voting machines and ballot scanners can suffer from analogous problems—miscalibrated touch-screens or dirty scanner heads can prevent votes in specific ballot positions from being recorded [31]—but they also have more complex failure modes that call for different forms of testing. These devices must be provisioned with an "election definition" that specifies the ballot layout and rules. If the election definition is wrong—for instance, the order or position of voting targets do not match the ballots a scanner will read—votes may be miscounted.

Problems with election definitions caused by human error are surprisingly common. They contributed to the publication of incorrect initial election results in Northampton County, Pennsylvania, in 2019 [14], Antrim County, Michigan, in 2020 [28], and DeKalb County, Georgia, in 2022 [23]. In these documented cases the errors were fortunately detected, but only after the results were announced. They likely could have been prevented in the first place by sufficient L&A testing.

L&A testing can also serve a role in election security. Research has long recognized that L&A testing *cannot* reliably defeat an adversary who manages to execute malware on voting machines, because the malware could detect when it was under test and only begin cheating during the election itself (see, e.g., [24]).

However, L&A testing can potentially thwart more limited attackers who manage to tamper with election definitions or configuration settings. For example, although there is no evidence that the instances of error described above were caused by fraud, attackers could cause similar election definition problems deliberately in an attempt to alter results. This would likely require far less sophistication than creating vote-stealing malware. Moreover, there is growing concern about threats posed by dishonest election insiders, who routinely have the access necessary to perform such an attack [17].

Beyond providing these protections, L&A testing also frequently serves a role in enhancing public confidence in elections. Most states conduct at least part of their L&A testing during a public ceremony, where interested political party representatives, candidates, news media, and residents can observe the process and sometimes even participate by marking test ballots. Some jurisdictions also provide live or recorded video of their testing ceremonies online. These public tests can help build trust by allowing voters to meet their local officials, observe their level of diligence, and become more familiar with election processes. Additionally, public observers have the potential to make testing stronger, by providing an independent check that the required tests were completed and performed correctly. At least in principle, public observation could also help thwart attempts by dishonest officials to subvert L&A testing by skipping tests or ignoring errors.

2.2 U.S. Elections

L&A testing fills a role that is best understood with a view towards the broader context of election administration in the jurisdictions where it is practiced. In the U.S., many subjects are put to the voters, frequently all at once, and a single ballot might include contests ranging from the national presidency and congress to the state governor, legislature, and judges to the local mayor, city council, sheriff, and school board [13]. This means elections tend to involve many contests—typically around 20, although some jurisdictions have occasionally had nearly 100 [76]. There may also be several ballot variants within a single polling place to accommodate candidates from different sets of districts. These features make tallying by hand impracticable in many areas. As a result, nearly all jurisdictions rely on electronic tabulation equipment, today most commonly in the form of computerized ballot scanners [70]. Ensuring that these machines are properly configured and functioning on election day is the key motivation for L&A testing.

Election administration in the U.S. is largely the province of state and local governments. Although the Constitution gives Congress the power to override state law regarding the "manner of holding Elections for Senators and Representatives," this authority has been applied only sparingly, for instance to establish accessibility requirements and enforce civil rights [30,73]. Each state legislature establishes its own election laws, and the state executive (typically the secretary of state) promulgates more detailed regulations and procedures. In practice, election administration powers are exercised primarily by local jurisdictions, such as counties or cities and townships, where local officials (often elected officials called "clerks") are responsible for conducting elections [45].

Because of this structure, there is little standardization of election practices across the states, and L&A testing is no exception. Testing processes (and the ceremonies that accompany them) vary substantially between and within states. As we show, these variations have significant effects, both with respect to error-detection effectiveness and procedural transparency and intelligibility. Pessimistically, one can view this broad local discretion as a way for some jurisdictions to use lax practices with little accountability. We note, however, that it also grants many clerks the power to depart *upwards* from their states' mandatory procedures, achieving stronger protections than the law requires. This provides an opportunity for improved practices to see early and rapid adoption.

2.3 Related Work

Although L&A testing itself has so far received little research attention, there is extensive literature analyzing other aspects of election mechanics across states and countries, with the goal of informing policymaking and spreading best practices. For instance, past work has examined state practices and their impacts regarding post-election audits [68], voter registration list maintenance [10], voter identification requirements [16], online voter registration [79], election observation laws [27], the availability of universal vote-by-mail [67]. A far larger body of research exists comparing state practices in fields other than elections.

Despite the abundance of this work, we are the first (to our knowledge) to examine states' L&A testing practices in detail. A 2018 state-by-state report by the Center for American Progress [58] considered L&A testing among several other aspects of election security preparedness; however, it primarily focused on the narrow question of whether states required all equipment to be tested. To build upon this research, we consider many other policy choices that influence the effectiveness of L&A requirements and procedures.

3 Methodology

3.1 Data Collection

To gather information on states' practices, we began by collecting official documentation where publicly available, relying primarily on state legal codes, state election websites, and Internet search engines. If we could not locate sufficient information, we attempted to contact the state via email or by phone to supplement our understanding or ask for clarifications. We directed these inquiries to the state elections division's main contact point, as identified on its website.

State responses varied. While some states provided line by line answers to each of our questions, it was common for states to indicate that our criteria were more specific than what state resources dictated, pointing us instead to the same statues and documentation we had already examined, providing us with additional documentation that was still unresponsive to our questions, or replying in paragraphs that partially addressed some questions while completely disregarding others. In cases where we could not find evidence to support that a state satisfied certain criteria and the state did not provide supporting evidence upon request, we did not award the state any points for those criteria.

Upon finalizing our summary of each state's practices, we contacted officials again to provide an opportunity for them to complete or correct our understanding. Over the course of nine months, we communicated with all 50 states and received at least some feedback on our summaries from all but seven states—Iowa, New Jersey, New York, Rhode Island, Tennessee, Vermont, and Wisconsin. Our data and analysis are current as of July 2022.

3.2 Evaluation Criteria

To uniformly assess and compare states practices, we applied the following criteria and scoring methodology, which reflect attributes we consider important for maximizing the benefits of L&A testing in terms of accuracy and voter confidence. These criteria are non-exhaustive, but we believe they are sufficiently comprehensive to evaluate state procedures relative to one another. (Additional desirable testing properties are discussed in Sect. 5.) Note that our assessments do not necessarily reflect practice in each of a state's subdivisions, since local officials sometimes have authority to exceed state guidelines. To keep the analysis tractable, we instead focus on the *baseline* established by statewide requirements.

We developed two categories of criteria: *procedural criteria*, which encompass the existence of procedures, the scope of testing, and transparency; and *functional criteria*, which reflect whether the testing could reliably detect various kinds of errors and attacks. To facilitate quantitative comparisons, we assigned point values to each criterion, such that each category is worth a total of 10 points and the weights of specific items reflect our assessment of their relative importance.

Procedural Criteria

Rules and Transparency (5 points). To provide the strongest basis for trust, testing should meet or exceed published requirements and be conducted in public.

RT1 (1.5 pts): Procedures are specified in a detailed public document.

This captures the threshold matter of whether states have published L&A requirements. Detailed or step-by-step guidelines received full credit, and general laws or policies received half credit.

RT2 (1.0 pts): The document is readily available, e.g., via the state's website.

Making L&A procedures easily available helps inform the public and enables observers to assess tests they witness.[1]

[1] Even when procedures are public documents, they are not always readily accessible. One state, Delaware, instructed us that we would need to find a resident to file a Freedom of Information Act request before their procedures would be provided.

RT3 (1.5 pts): Some testing is open to the public, candidates/parties, journalists.
This tracks the potential for public L&A ceremonies to strengthen confidence.
RT4 (1.0 pts): Local jurisdictions have latitude to exceed baseline requirements.

Scope of Testing (5 points). A comprehensive approach to testing covers every ballot design across all the voting machines or scanners where they can be used.

ST1 (2.0 pts): All voting machines/scanners must be tested before each election.
ST2 (1.0 pts): All devices must be tested *at a public event* before each election.
ST3 (2.0 pts): All devices must be tested with every applicable ballot design.

Failing to test all machines or all ballot styles risks that localized problems will go undetected, so each was assigned a substantial 2 points. One additional point was provided if all testing is public, to reflect transparency interests.

Functional Criteria

In each of three sets of functional criteria, we assess a simple form of the protection (with a small point value) and a more rigorous form (with a large point value).

Basic Protections (4 points). To guard against common errors, tests should cover every voting target and ensure detection of transpositions.

BP1 (1.0 pts): All choices receive at least one valid vote during testing.
BP2 (3.0 pts): No two choices in a contest receive the same number of votes.
The first test minimally detects whether each candidate has some functioning voting target. The second further ensures the detection of transposed targets within a contest, which can result from misconfigured election definitions.

Overvote Protection (2 points). Testing should exercise overvote detection and, ideally, confirm that the overvote threshold in each contest is set correctly.

OP1 (0.5 pts): At least one overvoted ballot is cast during testing.
OP2 (1.5 pts): For each contest c, a test deck includes a ballot with n_c selections and one with n_c+1 selections, where n_c is the permitted number of selections.
An overvote occurs when the voter selects more than the permitted number of candidates, rendering the selections invalid. The first practice minimally detects that the machine is configured to reject overvotes, while the second tests that the allowed number of selections is set correctly for each contest.

Nondeterministic Testing (4 points). For stronger protection against deliberate errors, attackers should be unable to predict how the test deck is marked.

ND1 (1.0 pts): Public observers are allowed to arbitrarily mark and cast ballots.
ND2 (3.0 pts): Some ballots must be marked using a source of randomness.
Attackers who can predict the test deck potentially can tamper with the election definition such that errors will not be visible during testing. If the public can contribute test ballots, this introduces uncertainty for the attacker, while requiring random ballots allows for more rigorous probabilistic detection.

State	Procedural Criteria							
	RT1	RT2	RT3	RT4	ST1	ST2	ST3	∑
	1.5	1.0	1.5	1.0	2.0	1.0	2.0	(/10)
MT	●	●	●	●	●	●	●	10.00
OH	●	●	●	●	●	●	●	10.00
PA	●	●	●	●	●	●	●	10.00
WA	●	●	●	●	●	●	●	10.00
CO	◐	●	●	●	●	●	●	9.25
CT	◐	●	●	●	●	●	●	9.25
MA	◐	●	●	●	●	●	●	9.25
MO	◐	●	●	●	●	●	●	9.25
NV	◐	●	●	●	●	●	●	9.25
SD	◐	●	●	●	●	●	●	9.25
UT	◐	●	●	●	●	●	●	9.25
VA	●	●	◐	●	●	●	●	9.25
WV	◐	●	●	●	●	●	●	9.25
WI	◐	●	●	●	●	●	●	9.25
WY	◐	●	●	●	●	●	●	9.25
MI	●	●	●	●	●	○	●	9.00
MN	●	●	●	●	●	○	●	9.00
NH	●	●	●	⊙	●	●	●	9.00
VT	●	●	●	⊙	●	●	●	9.00
ID	◐	◐	●	●	●	●	●	8.75
AK	●	◐	●	⊙	●	●	●	8.50
DE	●	◐	●	⊙	●	●	●	8.50
GA	●	◐	●	⊙	●	●	●	8.50
NC	●	◐	●	⊙	●	●	●	8.50
AL	◐	●	●	●	●	○	●	8.25
ND	◐	●	●	⊙	●	●	●	8.25
AZ	●	●	●	●	●	●	⊙	8.00
IL	●	◐	●	●	●	●	○	7.50
FL	●	●	●	●	●	○	○	7.00
IN	●	●	●	●	○	○	●	7.00
SC	●	◐	●	⊙	●	●	⊙	6.50
IA	◐	●	●	⊙	●	●	⊙	6.25
KS	◐	●	●	⊙	●	●	⊙	6.25
MS	◐	●	●	⊙	●	●	⊙	6.25
RI	◐	●	●	⊙	●	●	⊙	6.25
TX	◐	●	●	●	○	○	●	6.25
AR	●	●	●	⊙	●	⊙	⊙	6.00
NE	●	◐	⊙	⊙	●	⊙	●	6.00
KY	◐	●	○	⊙	●	○	●	5.75
MD	◐	●	○	⊙	●	○	●	5.75
NY	◐	●	⊙	⊙	●	⊙	●	5.75
CA	◐	●	◐	●	●	○	○	5.50
LA	◐	●	◐	⊙	●	●	⊙	5.50
NM	◐	●	◐	○	●	●	○	5.50
HI	●	◐	○	●	●	○	○	5.00
OR	◐	●	◐	⊙	●	○	⊙	4.50
ME	◐	●	●	⊙	⊙	⊙	⊙	3.25
NJ	◐	●	●	⊙	⊙	⊙	⊙	3.25
OK	●	◐	◐	⊙	⊙	⊙	⊙	2.75
TN	◐	●	◐	⊙	○	○	⊙	2.50

State	Functional Criteria						
	BP1	BP2	OP1	OP2	ND1	ND2	∑
	1.0	3.0	0.5	1.5	1.0	3.0	(/10)
CT	●	●	●	●	○	◐	7.50
AZ	●	⊙	●	●	●	●	7.00
SD	●	●	●	●	●	○	7.00
IL	●	●	●	●	○	○	6.00
MO	●	●	●	●	○	○	6.00
NE	●	●	●	●	⊙	⊙	6.00
MI	●	●	●	○	●	○	5.50
UT	●	●	●	○	●	○	5.50
VT	●	●	⊙	⊙	⊙	◐	5.50
WY	●	●	●	○	●	○	5.50
AR	●	●	●	⊙	⊙	⊙	4.50
DE	●	●	●	○	○	○	4.50
FL	●	●	●	⊙	⊙	○	4.50
MN	●	●	●	○	⊙	⊙	4.50
MT	●	●	●	○	○	○	4.50
ND	●	●	●	○	○	○	4.50
OH	●	●	●	⊙	⊙	⊙	4.50
WA	●	●	●	○	○	○	4.50
ID	●	○	●	●	●	○	4.00
NM	●	○	●	●	○	○	3.00
CO	●	○	●	○	●	○	2.50
IA	●	⊙	●	⊙	●	⊙	2.50
AL	●	⊙	●	⊙	⊙	⊙	1.50
GA	●	⊙	●	⊙	○	⊙	1.50
IN	●	⊙	●	⊙	○	○	1.50
KS	●	⊙	●	⊙	⊙	⊙	1.50
KY	●	⊙	●	⊙	⊙	⊙	1.50
ME	●	⊙	●	⊙	○	⊙	1.50
NV	●	⊙	●	⊙	⊙	⊙	1.50
NH	●	⊙	●	⊙	⊙	⊙	1.50
NJ	●	⊙	●	⊙	○	⊙	1.50
NY	●	⊙	●	⊙	⊙	⊙	1.50
PA	●	○	●	⊙	○	⊙	1.50
TN	●	⊙	●	⊙	⊙	⊙	1.50
TX	●	○	●	○	○	○	1.50
VA	●	⊙	●	⊙	⊙	⊙	1.50
WV	●	⊙	●	⊙	⊙	⊙	1.50
WI	●	⊙	●	⊙	⊙	○	1.50
HI	○	○	●	○	◐	○	1.00
MD	●	○	○	○	○	○	1.00
SC	●	⊙	○	⊙	⊙	⊙	1.00
MA	○	○	●	○	○	○	0.50
MS	⊙	⊙	●	⊙	○	⊙	0.50
NC	⊙	⊙	●	⊙	○	⊙	0.50
OK	⊙	⊙	●	⊙	○	⊙	0.50
AK	⊙	⊙	⊙	⊙	⊙	⊙	0.00
CA	○	○	○	○	○	○	0.00
LA	⊙	⊙	⊙	⊙	○	⊙	0.00
OR	⊙	⊙	⊙	⊙	○	⊙	0.00
RI	⊙	⊙	⊙	⊙	⊙	⊙	0.00

●: Fully met ◐: Partly met ○: Not met ⊙: State not responsive (scored as unmet)

4 Analysis

Our nationwide review of L&A procedures highlights significant variation among the testing practices of the fifty states, as illustrated by the maps in Fig. 2. The tables on page 7 summarize our findings and rank the states with respect to the procedural and functional criteria. We also provide a capsule summary of each state's practices in Appendix A.

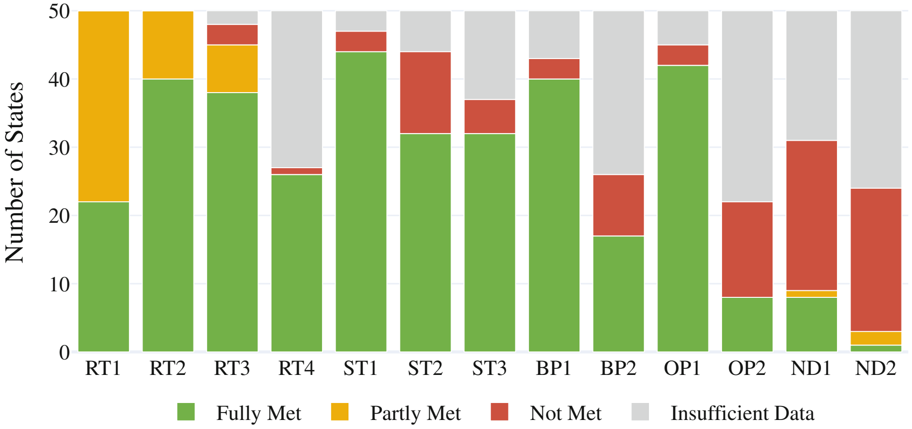


Fig. 1. We count the number of states that met, partly met, or did not meet each criterion. While states commonly require simple protections (BP1, OP1), most do not achieve more rigorous forms of error detection (e.g., OP2, ND1, and ND2).

4.1 Performance by Criterion

Figure 1 shows the number of states that met, partly met, or did not meet each criterion. All 50 states have laws that require L&A testing, but only 22 have a public, statewide document that details the steps necessary to properly conduct the tests (RT1). Of those that do not, several (such as California) merely instruct local jurisdictions to follow instructions from their voting equipment vendors, limiting the efficacy of logic and accuracy procedures to each vendor's preferences.

States generally performed well with respect to transparency criteria. Every state has some public documentation about its L&A practices, with 40 states making this documentation readily available (RT2). At least 45 states perform some or all testing in public (RT3), although 7 of these impose restrictions on who is allowed to attend. At least 32 states test every machine in public (ST2). Just three states (Kentucky, Maryland, and Hawaii) do not conduct any public L&A testing, which may be a significant lost opportunity to build public trust.

Most states also scored high marks regarding the scope of their testing. We were able to confirm that at least 44 states require all equipment to be tested before each election (ST1). Exceptions include Tennessee, Texas, and Indiana, which only require testing a sample of machines. At least 32 states require every ballot style to be tested, but 5 or more do not (ST3), which increases the chances that problems will go undetected in these jurisdictions.

Consideration of several other criteria is more complicated, because we often lack evidence for or against their being met. We have insufficient data about 19 or more states for RT4 and most of the functional criteria (BP2, OP2, ND1, and ND2). Details concerning functional criteria tended to be less frequently described in public documentation, which potentially biases our analysis in cases where states were also unresponsive to inquiries. We treated such instances as unmet for scoring purposes, but it is also informative to consider the ratio of met to unmet, as depicted in Fig. 1. One example is whether states allow local jurisdictions to exceed their baseline requirements (RT4). Although we lack data for 23 states, the criterion is met by at least 26 states, and we have only confirmed that it is unmet in one state (New Mexico). This suggests that many of the unconfirmed states likely also allow local officials to depart upwards from their requirements.

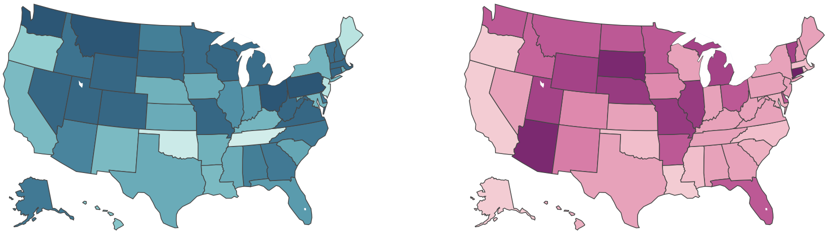
After accounting for what data was available, states clearly perform much better on our procedural criteria than on our functional criteria. This suggests that many of the functional attributes we looked for are aspirational relative to current practice and indicates that L&A testing *could* provide much more value.

The only two functional criteria that most states meet are basic protections for voting targets (BP1) and overvotes (OP1), which are provided in at least 40 and 42 states, respectively. At least 17 states would detect transposed voting targets within contests (BP2), but as few as 8 fully validate overvote thresholds (OP2). These more rigorous protections require more complicated procedures and larger test decks, but that some states achieve them suggests that they would be practical to implement more broadly.

Policies facilitating even the basic form of nondeterministic testing were rare. Only 11 states scored even partial points for conducting nondeterministic testing, with 9 of them allowing public observers to mark test ballots (ND1) and 3 of them (Arizona, Connecticut, Vermont) confirming that election officials are required to mark random selections (ND2). Of the three, only Arizona confirmed that it required officials to use a random number generator, thus earning full points. These findings are surprising, since unpredictable or randomized testing can thwart certain kinds of attacks that predictable tests cannot. That nondeterministic testing is rare greatly limits the security benefits of typical state L&A practices.

4.2 Performance by State

When comparing states' overall L&A testing practices, we find wide variation across both procedural and functional criteria. As illustrated in Fig. 2, this vari-



(a) Procedural Score (b) Functional Score

Fig. 2. Mapping state scores (darker indicates better performance) shows that L&A testing practices vary significantly within all geographic regions of the U.S.

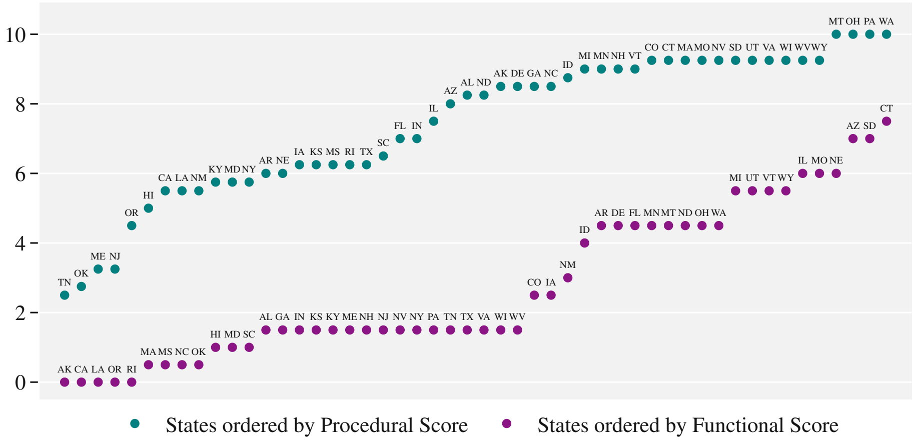


Fig. 3. Many states have perfect or near-perfect procedural scores, but functional scores are generally lower, reflecting opportunities for making L&A more effective.

ation is not clearly explained by regionalism. However, the plot in Fig. 3 reveals several notable features in the distributions of states' scores.

Most obviously, procedural scores were much higher than functional scores. Again, this likely reflects both the relative scarcity of public documentation about functional aspects of L&A testing and that our chosen functional criteria were somewhat aspirational. No states achieved perfect functional scores, but 4 states (Montana, Ohio, Pennsylvania, and Washington) achieved perfect procedural scores. Four other states could potentially achieve this benchmark but did not provide missing information we requested. Eleven more states clustered just shy of perfect procedural scores, of which 10 could achieve full points simply by making detailed L&A procedures public (BP1)—potentially a zero-cost policy change.

Notable relationships occur between certain criteria. For instance, concerning the scope of testing, only 2 states that are known to require testing every ballot

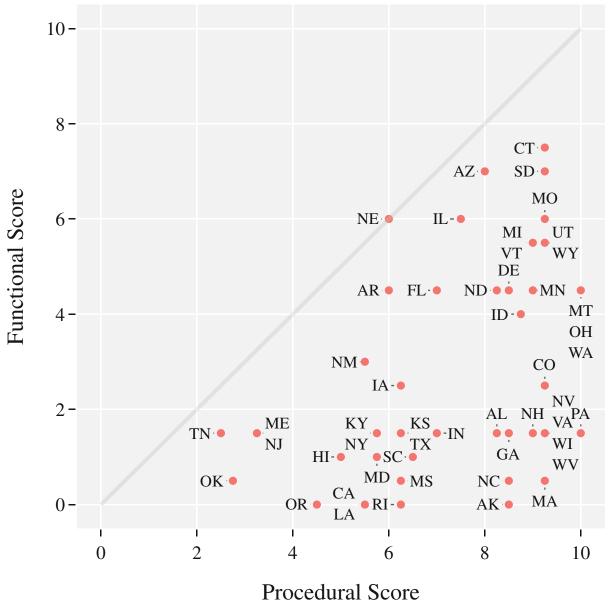


Fig. 4. No state's functional score exceeds its procedural score, perhaps due to more limited data about functional aspects of testing. At most levels of procedural scores, states' functional scores spanned a wide range, with no strong correlation.

(ST3) style do not also require testing every machine (ST1). It is much more common for states that require testing every machine to not require testing every ballot style (or remain silent), which 14 of 44 states did. This suggests that L&A policymakers are more likely to be aware of the potential for problems that affect only specific machines than of issues that can affect only specific ballot styles.

The distribution of functional scores highlights further relationships. The largest cluster, at 1.5, are 16 states that require basic protections for voting targets (BP1) and overvotes (OP1) but meet no other functional criteria. Interestingly, many of these states employ similar statutory language, with little or no variation.[2] Although we have so far been unable to identify the common source these states drew on, the situation suggests that providing stronger model legislation could be a fruitful way to encourage the adoption of improved L&A practices.

At least 21 other states accomplish basic voting-target and overvote protections plus one or more of the stronger functional criteria. Most commonly, they require additional testing to detect transposed voting targets within contests (BP2), which 17 states do. Eight of these states accomplish no further functional criteria, resulting in a cluster at score 4.5. Five others also fully validate overvote thresholds (OP2), as do only 3 other states, indicating a strong correlation

[2] An additional state, South Carolina, adopted nearly the same statutory formula but with a small change that *weakens* overvote protection, and so does not satisfy OP1.

between these more rigorous testing policies. In a surprising contrast, although nondeterministic testing is comparably uncommon (only 8 states fully achieve either ND1 or ND2), practicing it does *not* appear to be well correlated any of the other non-basic functional criteria. This may indicate that states have introduced nondeterministic testing haphazardly, rather than as the result of careful test-process design.

Considering both scoring categories together (Fig. 4), we see that although no state's functional score exceeds its procedural score, there is otherwise a wide range of functional scores at almost every level of procedural score. This may partly reflect limitations due to unresponsive states, but it may also suggest opportunities to better inform policymakers about ways to strengthen L&A functionality, particularly in states in the lower-right corner of the figure, which have a demonstrated ability to develop procedurally robust testing requirements.

The overall national picture shows that every one of our evaluation criteria is satisfied by at least one state, indicating that even the most rigorous functional criteria are realistic to implement in practice. Several states have the potential to serve as models of best practice across most dimensions of L&A testing, especially if procedural specifics are made readily accessible. In particular, Arizona and South Dakota each achieved full points in all but one criterion from each category, and Connecticut achieved the highest total score of any state under our metrics. We provide additional information and references regarding their procedures in our state-by-state summaries, found in Appendix A.

5 Discussion

Our findings support the need for strengthened L&A procedures nationwide. Current practice has room for substantial improvement in both transparency and substance, and state policy should seek to realize this potential.

Election security researchers and practitioners should work together to establish normative standards for L&A testing procedures and to draft model legislation to realize them. The precise mechanism for establishing this standard is beyond the scope of this paper, but a potential route would be for the National Institute of Standards and Technology (NIST) to issue L&A testing guidelines. Under the Help America Vote Act (HAVA), NIST is charged with the design of voting system standards, in coordination with the U.S. Election Assistance Commission (EAC) [30], and it has previously issued guidance for other aspects of election technology administration, such as cybersecurity and accessibility.

One challenge in the adoption of any technical standard is leaving safe and flexible opportunities for upward departure. It would be dangerous to lock in procedures that are later found to be insufficient, especially if every state would have to update its laws in response. For this reason, it is important that any L&A policy changes allow some degree of flexibility involved for local jurisdictions. Too much flexibility, however, can weaken security guarantees even with the best of intentions. One clerk we spoke with in the preparation of this paper offhandedly told to us that she did not always follow the state requirement that every candidate in a contest receive a different number of votes, since in real elections ties could occur and she felt it was important to test that behavior too. Despite the well-meaning nature of this deviation, it *decreased* the guarantees provided by her L&A testing, since it meant the two candidates who tied in the test deck could have had their votes unnoticeably swapped. Clerks do not have the resources to rigorously analyze all ramifications of deviations from procedure, so latitude to deviate should be provided only where it cannot reduce the integrity of the process, such as in optional, additional phases of testing.

We leave to future work determining what model L&A policies should look like. While the elements of transparency, openness, and security we considered in this paper are potential low-hanging fruit, there are other elements of successful L&A practice that we did not measure or describe. For instance, testing policies should consider not only ballot scanners but also ballot marking devices (BMDs), which are computer kiosks that some voters use to mark and print their ballots. Most jurisdictions use BMDs primarily for voters with assistive needs, but some states require all in-person voters to use them [70]. Errors in BMD election definitions can lead to inaccurate results [23], but carefully designed L&A testing might reduce the incidence of such problems. Another example of an intervention that would have detected real-world issues in the past is "end-to-end" L&A testing, where tabulator memory cards are loaded into the central election management system (EMS) and its result reports are checked against the test decks. One of the problems in Antrim County that caused it to report initially incorrect results in 2020 was an inconsistency between some tabulators and the EMS software, and "end-to-end" L&A testing could have headed off this issue [28].

We do, however, recommend that future L&A guidelines incorporate elements of nondeterministic testing. While our data shows that this practice is still quite rare in the U.S., using test decks that are unpredictable would make it more difficult to construct malicious election definitions that pass the testing procedure.

Election technology has evolved over time, but some L&A testing practices still carry baggage from the past. For instance, functional requirements in many U.S. states are suited for detecting common problems with mechanical lever voting machines but less adept at uncovering common failure modes in modern computerized optical scanners, such as transposed voting targets. Other nations, which may at this time be adopting optical scan equipment of their own, can learn from these standards and improve on them as they choose their own practices for the future. By applying careful scrutiny of existing process and incorporating the elements that most make sense in their own context, these polities can ensure their testing procedures are constructed in a way to meet their needs.

6 Conclusion

We performed the first detailed comparative analysis of L&A testing procedures, based on a review of L&A requirements and processes across all fifty U.S. states.

Although L&A testing can be a valuable tool for spotting common configuration errors and even certain kinds of low-tech attacks, our analysis shows that there is wide variation in how well states' testing requirements fulfill these prospects. We hope that our work can help rectify this by highlighting best practices as well as opportunities for improvement. Rigorous, transparent L&A testing could also be a valuable tool for increasing public trust in elections, by giving voters a stronger basis for confidence that their votes will be counted correctly.

Acknowledgements. We are grateful to the many election workers who corresponded with us to provide data for this study. We also thank our shepherd Nicole Goodman as well as Doug Jones, Dhanya Narayanan, Mike Specter, Drew Springall, and the anonymous reviewers. This work was supported by the Andrew Carnegie Fellowship, the U.S. National Science Foundation under grant CNS-1518888, and a gift from Microsoft.

A State-by-State Practices

Here we summarize notable features of each state's L&A testing practices with respect to our evaluation criteria. We list each state's score and rank under the procedural and functional criteria (each /10 points) and their total (/20 points).

Alabama (AL) *Proc.*: 8.25 (25th) *Func.*: 1.5 (23rd) *Total*: 9.75 (28th)

Alabama's L&A [5] public testing requirements vary by class of device. Direct-recording electronic voting devices (DREs), which are no longer used [70], had to be tested at an event open to the general public. Testing of the state's current optical scan machines occurs in two phases: all devices are tested in a process observable by candidates or their representatives, and a subset of the devices is tested again in view of the general public. Interestingly, the state requires each candidate receive a minimum of *two* votes, although there is no requirement that candidates within a contest receive a *different* number of votes. Alabama permits local jurisdictions to implement more stringent testing practices in addition to required testing, so practices could be independently improved by local jurisdictions.

Alaska (AK) *Proc.*: 8.5 (21st) *Func.*: 0.0 (46th) *Total*: 8.5 (32nd)

Documents describing Alaska's L&A procedures are not publicly accessible, but the state's Division of Elections provided them in heavily redacted form in response to our requests [6–9]. Although Alaska requires each voting machine to be publicly tested, the redactions precluded our finding evidence responsive to most of our other criteria, resulting in a low score. Alaska has not responded to a request for clarification.

Arizona (AZ) *Proc.*: 8.0 (27th) *Func.*: 7.0 (2nd) *Total*: 15.0 (4th)

Arizona uses a random number generator to mark test ballots, creating a non-deterministic test deck that would be difficult to predict and earning the state full credit for ND2. The state also firmly bounds the overvote thresholds by voting at least one ballot for the maximum number of permissible choices in each contest and then exactly one additional vote than what is allowable [11]. Lastly, the state permits observers from local political parties to contribute their own votes to a test ballot. The state could earn a perfect score overall by testing each ballot style and ensuring candidates within a contest receive distinct vote totals.

Arkansas (AR) *Proc.*: 6.0 (37th) *Func.*: 4.5 (11th) *Total*: 10.5 (25th)

Arkansas allows the public to observe the testing of all devices prior to an election. The state ensures every candidate receives at least one vote and that at least one overvoted ballot must be cast [12]. Its procedures can be strengthened by specifying proper overvote validation practice, ensuring different vote totals for each candidate in a contest, and introducing nondeterministic test elements. Arkansas has not responded to requests for additional information.

California (CA) *Proc.*: 5.5 (42nd) *Func.*: 0.0 (46th) *Total*: 5.5 (44th)

California's Election Code §15000 [18] requires the state to conduct logic and accuracy testing. The state, however, relies on vendor-provided procedures that vary from machine to machine instead of implementing statewide requirements that are independent of the vendor. We assessed that this reliance on vendor-provided testing material does not satisfy our functional criteria, since even if every vendor's manual happens to meet a particular requirement now, updated manuals could weaken these provisions without any conscious regulatory action. Accordingly, the state scores lower than it likely would have otherwise.

Colorado (CO) *Proc.*: 9.25 (5th) *Func.*: 2.5 (21st) *Total*: 11.75 (18th)

Colorado has robust procedural L&A policies that are publicly applied to every machine [20]. Still, Colorado has room to improve in their functional protections. The state currently does not require vote totals to differ between various candidates for an office, but instead encourages it by providing a "ballot position calculator." Furthermore, Colorado does not require a strict test of overvote protection. Most of the information for our assessment was provided via communication with the state as there is no statewide procedure document.

Connecticut (CT) *Proc.*: 9.25 (5th) *Func.*: 7.5 (1st) *Total*: 16.75 (1st)

Connecticut's public statues require L&A testing [21]. Communication with the state election division revealed that the state comes close to fulfilling almost all of our criteria and made it the highest-scoring state overall. The main opportunity we find for improvement would be to strengthen nondeterministic testing. Connecticut already earns partial credit for ND2 since its procedures demonstrate an understanding of the importance of randomized testing, even though the source of randomness is not specified. To earn a perfect functional score, the state should require the use of a random number generator to mark some number of test ballots and also allow public observers to mark and cast test ballots arbitrarily.

Delaware (DE) *Proc.*: 8.5 (21st) *Func.*: 4.5 (11th) *Total*: 13.0 (14th)

Delaware's L&A policies are unclear based on its statutes [1], but a FOIA request for testing policies submitted at our behest by a Delaware resident yielded more informative documents [22]. We therefore consider criterion RT1 to be met, although the process of obtaining the documentation was needlessly difficult. In addition to a lack of transparency around their L&A documentation, it is unclear whether the state allows local jurisdictions to practice more stringent testing requirements. However, all other procedural testing criteria were met. In terms of functional protections, both basic protections were included as part of testing requirements. Overvote protections are not strictly checked; test decks include ballots where all choices are marked for each office but do not include ones marked with exactly one more vote than is permissible. Additionally, election officials are not required to mark test ballots randomly.

Florida (FL) *Proc.*: 7.0 (29th) *Func.*: 4.5 (11th) *Total*: 11.5 (19th)

Florida tests all tabulators during a “100% Logic and Accuracy Testing” event and a sample during a public L&A test [25]. The state meets all procedural testing criteria except for requiring all machines to be tested publicly and with each ballot style. Florida recognizes that some counties use a traditional 1-2-3 test deck pattern (satisfying BP2) but notes that it is not the most accurate way to verify that ballots are being counted correctly. To supplement this testing mechanism, the state encourages (although does not require) the creation of an “‘enhanced test deck with non-traditional vote patterns.”

Georgia (GA) *Proc.*: 8.5 (21st) *Func.*: 1.5 (23rd) *Total*: 10.0 (27th)

Georgia maintains a step-by-step document that describes how to implement its L&A rules [26]. The procedures, however, do not meet many criteria including requiring testing of all ballot styles and ensuring candidates for an office have different vote totals. Additionally, the procedures could not be located from the state's election website.

Hawaii (HI) *Proc.*: 5.0 (45th) *Func.*: 1.0 (39th) *Total*: 6.0 (43rd)

Hawaii's L&A procedures [29] differ substantially from most other states. The state does not permit the general public to observe testing. Instead, “Official Observers” must be designated by a political party, news media organization, or chief election officer to “serve as the ‘eyes and ears’ of the public.” In practice, “Official Observers” are quasi election officials who are tasked with conducting testing however they choose. Given Hawaii's unique situation, we assigned the state partial credit for ND1. However, since testing is not open to members of the general public unless they become “Official Observers,” Hawaii did not earn credit for RT3. Hawaii earned full credit for requiring all tabulators to be tested, but it failed to meet other key criteria including ensuring that each candidate receives at least one vote and that all ballot styles are included during testing.

Idaho (ID) *Proc.*: 8.75 (20th) *Func.*: 4.0 (19th) *Total*: 12.75 (15th)

Although Idaho does not have a statewide document dedicated to explaining the state's testing procedures, communication with state officials revealed that Idaho meets all functional criteria except for requiring proper overvote validation (BP2) and for election officials to mark ballots using a source of randomness (ND2).

Illinois (IL) *Proc.*: 7.5 (28th) *Func.*: 6.0 (4th) *Total*: 13.5 (12th)

The Illinois L&A testing best practice guide [33] fulfills many functional and procedural criteria. Importantly, the state tactfully juxtaposes an example unsatisfactory test with an example satisfactory test in way that efficiently conveys its L&A requirements—a method that states hoping to improve procedural clarity should consider. Illinois would benefit from introducing nondeterministic elements to its L&A testing and requiring that all ballot styles be tested on each tabulator.

Indiana (IN) *Proc.*: 7.0 (29th) *Func.*: 1.5 (23rd) *Total*: 8.5 (32nd)

Indiana distinguishes between public testing and logic and accuracy testing in its procedure manual [34], requiring the former and leaving the latter undefined. What the state terms "public testing," however, is analogous to other states' L&A. While the state requires testing of all optical scan ("ballot card") tabulators, only 5% of DRE voting systems must be tested. Since much of Indiana still uses DREs [70], it should strongly consider requiring all devices to undergo testing prior to each election as well as introducing explicit requirements regarding basic protections, overvote threshold validation, and nondeterminism. Indiana does require that all ballot styles be tested on each device subject to testing.

Iowa (IA) *Proc.*: 6.25 (32nd) *Func.*: 2.5 (21st) *Total*: 8.75 (31st)

Iowa has limited public documentation of its election procedures and has not responded to any of our email inquiries regarding its testing practices. Therefore, the state was scored in accordance with what could be found in its Election Code [32] and an election security informational video on its website [35].

Kansas (KS) *Proc.*: 6.25 (32nd) *Func.*: 1.5 (23rd) *Total*: 7.75 (35th)

Kansas includes L&A testing requirements as part of its state code, §25-4411 [36]. This statute requires public testing with a test deck that includes at least one overvoted ballot for each machine. Kansas also requires that L&A testing be open to the public and that all machines are tested. This gives the state a somewhat better procedural testing score than it would have had otherwise, but there are still many improvements that can be made, such as specifying a requirement for the testing of each ballot style and detection of transposed targets. The state's election division has not responded to our inquiries regarding testing requirements and so has been scored purely on the basis of public information.

Kentucky (KY) *Proc.*: 5.75 (39th) *Func.*: 1.5 (23rd) *Total*: 7.25 (38th)

Kentucky does not have any statewide procedural L&A document, but the state does have L&A requirements under its administrative code [3]. Kentucky earned points for requiring all of its voting equipment to undergo testing; however, the state does not require public L&A testing. Instead, it permits a representative from each political party and representatives of news media to be present at a "Public Examination" as the county elections board ensures: ballots are properly arranged, counters are set to zero, equipment is locked, and that the equipment's assigned serial number is recorded.

Louisiana (LA) *Proc.*: 5.5 (42nd) *Func.*: 0.0 (46th) *Total*: 5.5 (44th)

Louisiana Election Code [38] requires testing of every machine before each election, but it does not appear to establish any minimum standards for what this testing entails. Unusually, only Louisiana citizens are permitted to observe testing, so the state earns only partial credit for RT3. The state still earned full credit for ST2 because it ensures that the permitted public observers may witness the preparation and testing of each machine.

Maine (ME) *Proc.*: 3.25 (47th) *Func.*: 1.5 (23rd) *Total*: 4.75 (46th)

Found in 1.21–A M.R.S.A [2], Maine's L&A policy has several opportunities for improvement. It is unclear whether each tabulator is required to be tested, whether all ballot styles are included in testing, whether two candidates from the same contest can receive the same number of votes, and so forth. Observers are also not permitted to test ballots and machines themselves. Maine should consider producing publicly accessible procedure or guideline documents that expand upon its policy. Maine's Bureau of Corporations, Elections and Commission has not responded to our email inquiries regarding its testing procedures.

Maryland (MD) *Proc.*: 5.75 (39th) *Func.*: 1.0 (39th) *Total*: 6.75 (40th)

Described in Maryland COMAR (33.10.01.14–16) [19], the state's L&A testing consists of a "Pre-Election Test" of all tabulators. It is followed by a "Public Demonstration" that is limited to attendance by one representative of each political party and independent candidate. The public demonstration only consists of documentation completed during the pre-test and an overview of the testing process. Maryland should consider permitting the general public to observe "Pre-Election Testing" and creating a readily accessible statewide procedural document so that the public can make informed observations.

Massachusetts (MA) *Proc.*: 9.25 (5th) *Func.*: 0.5 (42nd) *Total*: 9.75 (28th)

Massachusetts does not have an approved statewide document regarding L&A procedures, but it does have regulations on the subject [4]. Communication with state officials uncovered that the state's internal guidelines meet most procedural testing criteria. However, the state performs poorly in terms of test functionality, where the only criterion it meets is basic overvote protection (OP1).

Michigan (MI) *Proc.*: 9.0 (16th) *Func.*: 5.5 (7th) *Total*: 14.5 (7th)

Michigan's L&A procedures are documented in a public manual [39] and consist of a preliminary accuracy test in which all tabulators and BMDs are tested, as well as a public test in which only a sample of tabulators are required to be tested. Each candidate in a contest must receive a different number of votes, and overvoted ballots must be cast. Additionally, observers can mark and cast test ballots. We recommend that Michigan further ensure that all machines undergo public testing, that overvote thresholds are fully validated, and that some ballots are marked using a source of randomness.

Minnesota (MN) *Proc.*: 9.0 (16th) *Func.*: 4.5 (11th) *Total*: 13.5 (12th)

The state's public L&A guidelines [40] identify two testing events—a preliminary test in which all ballot counters are required to be tested and a public accuracy test in which only a sample is tested. Minnesota includes step-by-step instructions for creating test deck spreadsheets and provides samples that jurisdiction can utilize. The state does well in ensuring that its guidelines protect against problems identified in the basic protections category. The state could benefit from introducing nondeterministic elements to its testing as well as requiring election jurisdictions to fully validate overvote thresholds.

Mississippi (MS) *Proc.*: 6.25 (32nd) *Func.*: 0.5 (42nd) *Total*: 6.75 (40th)

Mississippi's L&A requirements can be found in its state code [41], which only requires that each machine be publicly tested for basic overvote rejection (OP1). The state has not responded to an inquiry regarding other procedural and functional elements of its testing.

Missouri (MO) *Proc.*: 9.25 (5th) *Func.*: 6.0 (4th) *Total*: 15.25 (3rd)

Missouri's L&A policy is defined in 15 CSR 30–10.040 [42]. Testing ensures that each candidate for an office receives a distinct and nonzero number of votes. Missouri is one of the few states that require full validation of overvote thresholds, with a requirement as follows: "In situations where a voter can legally vote for more than one person for an office, at least one card shall be voted for the maximum number of allowable candidates; one card shall [then] be marked to have one more vote for each candidate or question than is allowable." This language could serve as a model for other states. We recommend introducing nondeterministic testing but commend the state for its unusually clear functional policies.

Montana (MT) *Proc.*: 10.0 (1st) *Func.*: 4.5 (11th) *Total*: 14.5 (7th)

Under Montana's detailed public L&A procedures [43], testing is divided into three dimensions—functional, diagnostic, and physical—and fulfills almost all of our criteria. Montana could strengthen its procedures by fully bounding overvote thresholds and introducing forms of nondeterministic testing.

Nebraska (NE) *Proc.*: 6.0 (37th) *Func.*: 6.0 (4th) *Total*: 12.0 (17th)

Nebraska's logic and accuracy test is commonly referred to as a "Mock Election." All tabulators undergo three independent tests using different test decks—one by the election official, one by the chief election commissioner, and one by the person who installed the election definition on the voting device. The state's Election Act [46] does not indicate that testing observation is open to members of the public; however, its test procedures meet all other criteria in the scope of testing, basic protections, and overvote protection categories. Nebraska could further improve its test functionality by requiring nondeterministic testing.

Nevada (NV) *Proc.*: 9.25 (5th) *Func.*: 1.5 (23rd) *Total*: 10.75 (21st)

In Nevada, L&A testing is primarily described by state law [47]. All tabulators must be tested before an election in a process that can be observed by the general public, and local jurisdictions are allowed to exceed state testing requirements. Nevada would benefit from creating a public, statewide guide that describes the steps necessary to properly conduct L&A testing.

New Hampshire (NH) *Proc.*: 9.0 (16th) *Func.*: 1.5 (23rd) *Total*: 10.5 (25th)
New Hampshire's *Election Procedure Manual* [48] meets all scope of testing requirements but otherwise does not address several key criteria, including the more rigorous of basic and overvote protections. Although the state broadly requires that election officials mark ballots with "as many combinations as possible," we deem this to fall short of satisfying our nondeterministic testing criteria.[3] Calling for some test ballots to be marked truly at random would strengthen this provision.

New Jersey (NJ) *Proc.*: 3.25 (47th) *Func.*: 1.5 (23rd) *Total*: 4.75 (46th)
New Jersey's L&A policy is described by state statutes 19:53A-8 [51]. Testing is conducted publicly and includes one vote for each candidate as well as the casting of overvoted ballots. State law was unclear regarding other key aspects of testing, including whether all machines are to be tested prior to each election. The state did not respond to additional requests for information.

New Mexico (NM) *Proc.*: 5.5 (42nd) *Func.*: 3.0 (20th) *Total*: 8.5 (32nd)
New Mexico has a handbook of relevant election code and legislation, but no procedure document [49]. While this legislation requires public testing before each election, no other information was present relevant to our criteria. Further communication with state officials revealed that only "party and organization representatives, election observers and candidates" are allowed to observe testing. We did not receive a response on whether all ballot styles are tested on each tabulator. Functionally, the state's testing implements both overvote protections but only the first basic protection (BP1).

New York (NY) *Proc.*: 5.75 (39th) *Func.*: 1.5 (23rd) *Total*: 7.25 (38th)
New York has a relatively minimal set of requirements defining their L&A procedure [50], which they term "Prequalification Testing." Under these rules, all tabulators and all ballot styles are tested, but other important properties are not met: testing is not open to the public, overvote thresholds are not well-bounded, and multiple candidates in the same contest may receive an equal number of votes. We recommend addressing these shortfalls and introducing nondeterministic testing elements. The New York State Board of Elections did not respond to an email inquiry requesting additional information.

North Carolina (NC) *Proc.*: 8.5 (21st) *Func.*: 0.5 (42nd) *Total*: 9.0 (30th)
North Carolina's L&A testing is briefly described on the state's election website [52]; we were able to obtain the state's procedures via correspondence with the State Board of Elections. The step-by-step list meets most of our procedural criteria but almost none of our functional criteria.

North Dakota (ND) *Proc.*: 8.25 (25th) *Func.*: 4.5 (11th) *Total*: 12.75 (15th)
Email communication with state election officials provided a great deal of insight

[3] Taken literally, this is intractable. There are 2^n ways to mark n voting targets; an election with 50 candidates would require approximately as many ballots as there are grains of sand on earth.

regarding North Dakota's L&A testing [44]. The state met the more advanced criteria requiring that all ballot styles be tested on every machine (ST3) and that a different numbers of votes be assigned to each option for each contest. It does not, however, require validating overvote thresholds or any nondeterministic testing.

Ohio (OH) *Proc.*: 10.0 (1st) *Func.*: 4.5 (11th) *Total*: 14.5 (7th)

Ohio's L&A practices [53] indicate that state election officials are required to test tabulating computer programs prior to a given election and tabulating equipment prior to its use to count ballots. In addition to fulfilling each scope of testing and basic protections criterion, the state met all of our procedural criteria. The state would earn a perfect combined scored if it were to specify a requirements that election officials fully validate overvote thresholds and introduce nondeterministic elements into the test process.

Oklahoma (OK) *Proc.*: 2.75 (49th) *Func.*: 0.5 (42nd) *Total*: 3.25 (50th)

State law in Oklahoma contains some provisions related to testing but leaves significant gaps relative to our criteria. When we contacted the State Election Board to request more information, they provided a page from the state's *Uniform Election Reference* [54]. This still failed to answer many of our questions, but contained some language which suggests local election officials may have access to additional private documents further defining L&A requirements. The state did not answer follow-up requests for additional information, so we were only able to use public information and the short excerpt of the testing documentation when scoring the state's practice.

Oregon (OR) *Proc.*: 4.5 (46th) *Func.*: 0.0 (46th) *Total*: 4.5 (48th)

Oregon conducts all elections by mail and so defines L&A policy in its *Vote by Mail Procedure Manual* [55]. The state's L&A testing is divided into a preparatory test in which all tabulators are required to be tested and a public certification test in which only a sample is tested. Observation of public testing is limited to one representative of each party and each nonpartisan candidate or their designated representative. There is no indication that any of the criteria in basic protections, overvote protection, and nondeterminism are met.

Pennsylvania (PA) *Proc.*: 10.0 (1st) *Func.*: 1.5 (23rd) *Total*: 11.5 (19th)

Pennsylvania has robust testing requirements [56] with an appropriate scope and excellent transparency. The state also has several good functional recommendations, including encouraging jurisdictions to assign a different number of votes to each candidate in a contest. We recommend that Pennsylvania turn its recommended practices into requirements. The state should also introduce elements of nondeterminism into its testing practices.

Rhode Island (RI) *Proc.*: 6.25 (32nd) *Func.*: 0.0 (46th) *Total*: 6.25 (42nd)

Rhode Island's State Board of Elections works in conjunction with the voting equipment vendor to publicly test all tabulators before each election. Its L&A policy [57], however, is very vague regarding our functional criteria, which led to the state earning no credit for that category. The Board did not respond to our correspondence seeking additional information.

South Carolina (SC) *Proc.*: 6.5 (31st) *Func.*: 1.0 (39th) *Total*: 7.5 (37th)

Even though South Carolina's State Election Commission provided us with an excerpt of the state's L&A procedures [60], the document was so heavily redacted that we were not able to obtain any useful information relative to our criteria. Instead, we relied on state code [59], which uses similar language to several other states but is unique in that it can be satisfied by the inclusion of overvoted *or* undervoted ballots and thus fails to meet OP1. The State Election Commission did not responded to our request for additional information.

South Dakota (SD) *Proc.*: 9.25 (5th) *Func.*: 7.0 (2nd) *Total*: 16.25 (2nd)

South Dakota does not have an "approved" statewide document for conducting L&A, instead relying on its statutes [61,62]. Communication with state election officials revealed strong functional requirements. Notably, the state ensures proper overvote validation by requiring, for each contest, that election officials mark the maximum number of allowable votes on a test ballot and then exactly one more vote than what is permissible on another. South Dakota would benefit from requiring election officials to mark test ballots using a source of randomness and by creating publicly-accessible documentation that details statewide requirements.

Tennessee (TN) *Proc.*: 2.5 (50th) *Func.*: 1.5 (23rd) *Total*: 4.0 (49th)

Tennessee's L&A policy can be found in state law [64,65]. It allows candidate, news media, and (depending on the type of election) political party representatives to observe testing, but not the general public. Notably, the state earned no credit for ST1, ST2, and ST3 because it only requires testing of tabulators in a number of precincts equal to at least 1% of the total number of precincts in an election. The state did not responded to requests for additional information.

Texas (TX) *Proc.*: 6.25 (32nd) *Func.*: 1.5 (23rd) *Total*: 7.75 (35th)

Texas is positioned to greatly strengthen its L&A testing [66] by explicitly requiring the implementation of more rigorous practices. For example, although the state requires testing prior to a given election, it does not require that every device be tested, leaving the devices that are not tested susceptible to preventable errors. The state also requires that every candidate receive at least one vote, but does not ensure that a different number of votes is assigned to each candidate in a contest, thus leaving the possibility of transposed targets untested. Texas does, however, position its policy as a baseline requirement, so local jurisdictions have latitude to perform more comprehensive testing.

Utah (UT) *Proc.*: 9.25 (5th) *Func.*: 5.5 (7th) *Total*: 14.75 (5th)

Although Utah's testing requirements are currently only set forth by statute [69], the state's election division is developing a best practices guide during the summer of 2022. Email correspondence revealed that the state already performs well, meeting most of our criteria in both procedural and functional categories. Utah could earn a perfect score by fully validating overvote thresholds and ensuring that some test ballots are marked using a source of randomness.

Vermont (VT) *Proc.*: 9.0 (16th) *Func.*: 5.5 (7th) *Total*: 14.5 (7th)

Vermont's most recent L&A procedures [71] were produced in June 2022 and require that public testing incorporate all ballot styles and at least one overvote. Notably, the state requires election officials to randomly fill ten ballots while keeping different vote counts for all candidates. Since the state demonstrated an awareness of the importance of marking test ballots in a way that makes the outcome less predictable, we awarded partial credit for ND2, even though the policy does not explicitly require use of a random number generator. Vermont's election division did not responded to a request for additional information.

Virginia (VA) *Proc.*: 9.25 (5th) *Func.*: 1.5 (23rd) *Total*: 10.75 (21st)

Virginia's L&A procedures are outlined in Chap. 4 of its *General Registrar and Electoral Board Handbook* [72]. The state requires each locality to test all of its voting equipment with each ballot style prior to an election. However, its testing does not ensure that transposed targets are detected nor does it ensure that overvote thresholds are fully validated. The test procedures also do not incorporate nondeterministic elements. In addition to addressing these functional issues, Virginia should consider permitting members of the public who are not representatives of a candidate or political party to observe testing.

Washington (WA) *Proc.*: 10.0 (1st) *Func.*: 4.5 (11th) *Total*: 14.5 (7th)

Washington State meets all our procedural requirements. It tests all machines twice, first at a pretest and then again at a public proceeding [74]. For all elections in Washington, voters are allowed to mark only one option. Proper overvote validation for jurisdictions in this state would therefore look like ensuring every contest has at least one ballot that votes for precisely two options. Incorporating this practice would be a simple way to strengthen the state's functional score.

West Virginia (WV) *Proc.*: 9.25 (5th) *Func.*: 1.5 (23rd) *Total*: 10.75 (21st)

West Virginia [75] meets almost all of our procedural criteria. It requires testing of all tabulators prior to an election, permits additional testing in local jurisdictions, and ensures that every ballot style is included in the test deck. The state could strength functional aspects of its requirements by incorporating greater protections offered by BP2 and OP2 as well as by providing nondeterminism.

Wisconsin (WI) *Proc.*: 9.25 (5th) *Func.*: 1.5 (23rd) *Total*: 10.75 (21st)

Wisconsin maintains a state L&A policy that provides for the public testing of all machines with all ballot styles before each election [77]. Functionally, the policy requires at least the simple forms of basic protections and overvote protections. There is also a non-binding recommendation to test "as many vote combinations as possible." We recommend strengthening this so that election officials are required to test a different number of votes for each candidate in a contest and to mark some ballots using a source of randomness. The Wisconsin Elections Commission did not responded to a request for additional information.

Wyoming (WY) *Proc.*: 9.25 (5th) *Func.*: 5.5 (7th) *Total*: 14.75 (5th)
Wyoming has a strong L&A testing policy [78] which, among other provisions, requires that each candidate in a receive a different number of votes. From additional email correspondence with Wyoming's election division, we were able to verify that all tabulators are publicly tested. Wyoming would benefit from consolidating its L&A requirements into one resource and incorporating a source of randomness for marking some test ballots.

References

1. 2 DE Code § 5523 (2014 through 146th Gen Ass) (2014). https://law.justia.com/codes/delaware/2014/title-15/chapter-55/section-5523/
2. 21-A Me Rev. stat. § 854 (2011). https://legislature.maine.gov/statutes/21-A/title21-Asec854.html
3. 31 KY Admin. Regs. § 2:020 (2021). https://apps.legislature.ky.gov/law/kar/titles/031/002/020/. Accessed 15 Dec 2021
4. 950 Mass. Reg. § 54.02 (2022). https://casetext.com/regulation/code-of-massachusetts-regulations/department-950-cmr-office-of-the-secretary-of-the-commonwealth/title-950-cmr-5400-voting-and-counting-procedures-for-electronic-voting-systems/section-5402-testing. Accessed 19 Apr 2022
5. Alabama Electronic Voting Committee: Chapter 307-X-1, Procedures For Electronic Vote Counting Systems (2002). https://www.alabamaadministrativecode.state.al.us/docs/evc/307-X-1.pdf. Accessed 31 Mar 2002
6. Alaska Division of Elections: Democracy Suite Election Check List: Regional Offices (2020). Obtained via private communication. Accessed 14 Sept 2020
7. Alaska Division of Elections: LAT Testing for Early Vote: Absentee Voting Tablet HP (2021). Obtained via private communication. Accessed 19 Mar 2021
8. Alaska Division of Elections: LAT Testing for Precinct Scanner and Voting Tablet HP (2021). Obtained via private communication. Accessed 1 Mar 2021
9. Alaska Division of Elections: LAT Testing for Voting Tablet VVPAT and Precinct Scanner (2021). Obtained via private communication. Accessed 19 Mar 2021
10. Ansolabehere, S., Hersh, E.: The quality of voter registration records: a state-by-state analysis (2010). https://www.vote.caltech.edu/reports/6
11. Arizona Secretary of State: Arizona Elections Procedures Manual (2019). https://azsos.gov/sites/default/files/2019_ELECTIONS_PROCEDURES_MANUAL_APPROVED.pdf. Accessed Dec 2019
12. Arkansas State Board of Election Commissioners: County Board of Election Commissioners Procedures Manual (2022). https://static.ark.org/eeuploads/elections/2022_CBEC_Manual_FINAL.pdf
13. Ballotpedia: Michigan official sample ballots (2020). https://ballotpedia.org/Michigan_official_sample_ballots,_2020
14. Bartlett, B.: Report on the analysis of voting machine issues in Northampton county, county of Northampton: office of the county executive (2019). https://www.northamptoncounty.org/CTYEXEC/Documents/121219%20Press%20Conference%20Recap.pdf. Accessed 12 Dec 2019
15. Bernhard, M., et al.: Public evidence from secret ballots. In: Krimmer, R., Volkamer, M., Braun Binder, N., Kersting, N., Pereira, O., Schürmann, C. (eds.) E-Vote-ID 2017. LNCS, vol. 10615, pp. 84–109. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68687-5_6

16. Bowler, S., Donovan, T.: A partisan model of electoral reform: voter identification laws and confidence in state elections. State Polit. Policy Q. **16**(3), 340–361 (2016)
17. Brown, E., Gardner, A.: Georgia county under scrutiny after claim of post-election breach, The Washington Post (2022). https://www.washingtonpost.com/investigations/2022/05/13/coffee-county-misty-hampton-election/. Accessed 13 May 2022
18. California Elections Code, Division 15: Semifinal Official Canvass, Official Canvas, Recount, and Tie Vote Procedures; Chapter 2: Vote by Mail Ballot Processing. (2022) https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?lawCode=ELEC&division=15.&title=&part=&chapter=2.&article=. 1 Jan 2022
19. Code of Maryland Regulations: 33.10.01.14-16. https://www.dsd.state.md.us/COMAR/ComarHome.html
20. Colorado Secretary of State: C.R.S. Title 1 Elections, Article 7 Conduct of elections (2021). https://www.sos.state.co.us/pubs/info_center/laws/Title1/Title1Article7.html
21. CT Gen. Stat. § 9-247 (2012). https://law.justia.com/codes/connecticut/2012/title-9/chapter-147/section-9-247/
22. Delaware Department of Elections: Ballot Accuracy Procedures. https://www.muckrock.com/foi/delaware-236/ballot-accuracy-procedures-delaware-126706/?#files. Obtained via open records request
23. Estep, T.: DeKalb elections officials offer more insight into May primary issues. Atlanta J. Const. (2022). https://www.ajc.com/neighborhoods/dekalb/dekalb-elections-officials-offer-more-insight-into-may-primary-issues/KJPO7DK5KFD77KMQXXSWO2MXSA/. Accessed 14 July 2022
24. Feldman, A.J., Halderman, J.A., Felten, E.W.: Security analysis of the Diebold AccuVote-TS voting machine. In: USENIX Workshop on Electronic Voting Technology (EVT) (2007), August 2007
25. Florida Division of Elections. Guidelines for logic and accuracy (L&A) testing: DE reference guide 0019 (2020). https://soe.dos.state.fl.us/pdf/DE%20Guide%200019%20-%20Guidelines%20for%20Logic%20%20Accuracy%20Testing_rev20200709.pdf. Accessed July 2020
26. Georgia Secretary of State: Secure the Vote: Logic and Accuracy Procedures (2020). https://archive.org/download/logic-and-accuracy-procedures-v-1-02-2020-stv-1/Logic%20and%20Accuracy%20Procedures%20v1_02-2020%28STV%29%20%281%29.pdf. Accessed Jan 2020
27. Green, R.: Election observation post-2020. Fordham Law Rev. **90**(2) (2021)
28. Halderman, J.A.: The Antrim county 2020 election incident: an independent forensic investigation. In: 31st USENIX Security Symposium (2022)
29. Hawaii Office of Elections: Counting center manual (2020). https://elections.hawaii.gov/wp-content/uploads/2020-Counting-Center-Manual.pdf
30. Help America Vote Act of 2002, Pub. L. 107-252, 116 Stat. 1666
31. Hursti, H., Lindeman, M., Stark, P.B.: New Hampshire SB 43 Forensic Audit Report. (2021). https://www.doj.nh.gov/sb43/documents/20210713-sb43-forensic-audit-report.pdf. Accessed July 2021
32. IA Code Title 2: Elections and Official Duties (2022). https://www.legis.iowa.gov/law/iowaCode/chapters?title=II&year=2022
33. Illinois State Board of Elections, Division of Voting and Registration Systems: State of Illinois Voting System: Testing & Security Best Practices, Obtained via private communication
34. Indiana Election Division: Indiana Election Administrator's Manual. https://www.in.gov/sos/elections/files/2022-Election-Administrators-Manual.FINAL.pdf

35. Iowa Secretary of State: Election Security in Iowa (2018). https://sos.iowa.gov/electionsecurityiniowa.html
36. KS Stat §25-4411 (2016). https://daw.justia.com/codes/kansas/2016/chapter-25/article-44/section-25-4411
37. Lindeman, M., Stark, P.B.: A gentle introduction to risk-limiting audits. IEEE Secur. Priv. Spec. Issue Electron. Voting (2012). https://www.stat.berkeley.edu/~stark/Preprints/gentle12.pdf. Accessed 16 Mar 2012
38. Louisiana Sec. of State: State of Louisiana Election Code (2021). https://www.sos.la.gov/ElectionsAndVoting/PublishedDocuments/ElectionCode.pdf. Accessed Oct 2021
39. Michigan Department of State: Bureau of Elections: Test Procedure Manual For Tabulators and Voter Assist Terminals (VAT) (2019). https://www.michigan.gov/-/media/Project/Websites/sos/01vanderroest/TEST_DECK_MANUAL05.pdf
40. Minnesota Secretary of State: 2014 Minnesota Voting Equipment Testing Guide (2014). https://www.sos.state.mn.us/media/1108/2014-equipment-testing-guide.pdf
41. Mississippi Secretary of State: Mississippi County Election Handbook (2015). https://www.sos.ms.gov/content/documents/Elections/Mississippi%20%20County%20Election%20Handbook.pdf. Accessed Sept 2015
42. Mo. Code Regs. 15 §30–10.140 (2022). https://casetext.com/regulation/missouri-administrative-code/title-15-elected-officials/division-30-secretary-of-state/chapter-10-voting-machines-electronic/section-15-csr-30-10140-electronic-ballot-tabulation-counting-preparation-and-logic-and-accuracy-testing-dres-and-precinct-counters. Accessed 2 May 2022
43. Montana Secretary of State: Uniform Ballot and Voting Systems Procedures Guide: Security, Testing, Inventory, and Troubleshooting. https://sosmt.gov/wp-content/uploads/Uniform-Voting-Systems-Guide.pdf
44. N.D. Cent. Code § 16.1-06-15 (2011). https://casetext.com/statute/north-dakota-century-code/title-161-elections/chapter-161-06-ballots-voting-machines-electronic-voting-systems/section-161-06-15-mandatory-testing-of-electronic-voting-systems-before-each-election-and-after-tabulation-of-ballots. Accessed 1 Aug 2011
45. National Conference of State Legislatures: Election Administration at State and Local Levels (2020). https://www.ncsl.org/research/elections-and-campaigns/election-administration-at-state-and-local-levels.aspx. Accessed 3 Feb 2020
46. Nebraska Revised Statute 32-1049 (2007). https://nebraskalegislature.gov/laws/statutes.php?statute=32-1049
47. Nevada Revised Statutes: Title 24, Chapter 293B. https://casetext.com/statute/nevada-revised-statutes/title-24-elections/chapter-293b-mechanical-voting-systems-or-devices/testing-of-equipment-and-programs
48. New Hampshire Department of State: New Hampshire Election Procedure Manual: 2020–2021. https://sos.nh.gov/media/kzupydju/epm-2020.pdf
49. New Mexico Secretary of State: Election Handbook of the State of New Mexico (2021). https://www.sos.state.nm.us/wp-content/uploads/2021/10/NM-Election-Handbook-SOS.pdf
50. New York State Board of Elections: State of New York: 2021 Election Law (2021). https://www.elections.erie.gov/PDFs/2021ElectionLaw.pdf
51. NJ Rev. Stat. §19:53A-8 (2014) (1975). https://law.justia.com/codes/new-jersey/2014/title-19/section-19-53a-8/

52. North Carolina State Board of Elections: Preparing for Accurate Elections. https://www.ncsbe.gov/about-elections/election-security/preparing-accurate-elections
53. Ohio Secretary of State: Ohio Election Official Manual. https://www.sos.state.oh.us/globalassets/elections/directives/2022/eom/eom_fullversion_2022-02.pdf. Accessed 2 Feb 2022
54. Oklahoma State Election Board: Oklahoma Uniform Election Reference. https://archive.org/download/page-from-uniform-election-reference-200401/Pages%20from%20Uniform%20Election%20Reference%20200401.pdf. Obtained via open records request
55. Oregon Secretary of State: Vote by Mail Procedures Manual. https://sos.oregon.gov/elections/Documents/vbm_manual.pdf. Accessed Jan 2022
56. Pennsylvania Department of State: Directive on Logic & Accuracy Testing (2020). https://www.dos.pa.gov/VotingElections/OtherServicesEvents/Documents/PADOS_Directive_Logic_Accuracy%20with%20attestation.pdf
57. RI Gen. L. § 17-19-14 (2014) (1996). https://law.justia.com/codes/rhode-island/2014/title-17/chapter-17-19/section-17-19-14/
58. Root, D., Kennedy, L., Sozan, M., Parshall, J.: Election security in all 50 states, center for American progress report (2018). https://www.americanprogress.org/article/election-security-50-states/
59. SC Code §7-13-1390 (2019). https://www.scstatehouse.gov/code/t07c013.php
60. South Carolina Election Commission: South Carolina Election Preparation Results and Accumulation Guide (2019). https://archive.org/2/items/2021-10-18-sec-eprag-13-17-redacted/2021-10-18%20SEC%20EPRAG%2013-17_Redacted.pdf. Obtained via open records request
61. South Dakota Secretary of State: South Dakota Administrative Rule (2021). https://sdsos.gov/elections-voting/assets/2021ARSD.pdf. Accessed 6 Oct 2021
62. South Dakota Secretary of State: South Dakota Election Code (2021). https://sdsos.gov/elections-voting/assets/2021ElectionCode.pdf. Accessed 17 Aug 2021
63. Statutes of California: California Assembly Clerk Archive (1907). https://clerk.assembly.ca.gov/sites/clerk.assembly.ca.gov/files/archive/Statutes/1906_07/1907.pdf#page=691
64. Tenn. Code §29-9-105 (2020). https://law.justia.com/codes/tennessee/2020/title-29/chapter-9/section-29-9-105/
65. Tenn. Rules of Sec. of State, State Coordinator of Elections: Ch. 1360-2-1 (1999). https://publications.tnsosfiles.com/rules/1360/1360-02/1360-02-01.pdf
66. Tex. Elec. Code §1.001 (2021). https://statutes.capitol.texas.gov/Docs/SDocs/ELECTIONCODE.pdf
67. Thompson, D.M., Wu, J.A., Yoder, J., Hall, A.B.: Universal vote-by-mail has no impact on partisan turnout or vote share. Proc. Natl. Acad. Sci. **117**(25), 14052–14056 (2020). https://doi.org/10.1073/pnas.2007249117
68. U.S. Election Assistance Commision: Election Audits Across the United States (2021). https://www.eac.gov/sites/default/files/bestpractices/Election_Audits_Across_the_United_States.pdf. Accessed 6 Oct 2021
69. Utah State Code 20A-4-104: Counting ballots electronically (2022). https://le.utah.gov/xcode/Title20A/Chapter4/C20A-4-S104_2022050420220504.pdf. Accessed 4 May 2022
70. Verified Voting. The Verifier: Polling Place Equipment. https://www.verifiedvoting.org/verifier/

71. Vermont Secretary of State: Elections Division: Vermont Vote Tabulator Guide (2022). https://outside.vermont.gov/dept/sos/Elections%20Division/town%20clerks%20and%20local%20elections/election%20supplies/vermont-vote-tabulator-guide.pdf
72. Virginia Department of Elections: General Registrar and Electoral Board Handbook - Chapter 4 (2022). https://www.elections.virginia.gov/media/grebhandbook/2022-updates/4_Voting_Equipment_(2022).pdf
73. Voting Rights Act of 1965, Pub. L. 89-110, 79 Stat. 437
74. Washington Secretary of State: Presidential Primary Advisory 2020 #06 - Logic and Accuracy Testing (2020). https://www.sos.wa.gov/_assets/elections/06%20l&a%20testing_2020b.pdf. Accessed 21 Jan 2020
75. West Virginia Code: Ch. 3, Ar. 4, §26. https://code.wvlegislature.gov/3-4A-26/
76. Wheaton, S.: Comparing Ballot Lengths Around the Country, The New York Times (2013). https://archive.nytimes.com/thecaucus.blogs.nytimes.com/2013/02/06/comparing-ballot-lengths-around-the-country/?_r=0. Accessed 6 Feb 2013
77. Wisconsin Elections Commission: Election Administration Manual for Wisconsin Municipal Clerks (2020). https://elections.wi.gov/sites/elections/files/2022-03/Election%20Administration%20Manual%20%282020-09%29.pdf
78. Wyoming Admin. Rules: Sec. of Stat —Election Procedures—Ch. 29: Voting Systems (2020). https://rules.wyo.gov/DownloadFile.aspx?source_id=17964&source_type_id=81&doc_type_id=110&include_meta_data=Y&file_type=pdf&filename=17964.pdf&token=122222099183083056006234136082008020226028193044
79. Yu., J.: Does state online voter registration increase voter turnout? Soc. Sci. Q. **100**(3) (2019). https://onlinelibrary.wiley.com/doi/abs/10.1111/ssqu.12598

Author Index