Ceremonies for End-to-End Verifiable Elections

Aggelos Kiayias*[1], Thomas Zacharias†[1], and Bingsheng Zhang†‡[2]

[1]Dept. of Informatics and Telecommunications, University of Athens, Greece
[2]Security Lancaster Research Centre, Lancaster University, UK

{aggelos@di.uoa.gr, thzacharias@di.uoa.gr, b.zhang2@lancaster.ac.uk}

Abstract

State-of-the-art e-voting systems rely on voters to perform certain actions to ensure that the election authorities are not manipulating the election result. This so-called "end-to-end (E2E) verifiability" is the hallmark of current e-voting protocols; nevertheless, thorough analysis of current systems is still far from being complete.

In this work, we initiate the study of e-voting protocols as *ceremonies*. A ceremony, as introduced by Ellison [Ell07], is an extension of the notion of a protocol that includes human participants as separate nodes of the system that should be taken into account when performing the security analysis. We propose a model for secure e-voting ceremonies that centers on the two properties of end-to-end verifiability and privacy/receipt-freeness and allows the consideration of arbitrary behavioral distributions for the human participants.

We then analyze the Helios system as an e-voting ceremony. Security in the e-voting ceremony model requires the specification of a class of human behaviors with respect to which the security properties can be preserved. We show how end-to-end verifiability is sensitive to human behavior in the protocol by characterizing the set of behaviors under which the security can be preserved and also showing explicit scenarios where it fails.

We then provide experimental evaluation with human subjects from two different sources where people used Helios: the elections of the International Association for Cryptologic Research (IACR) and a poll of senior year computer science students. We report on the auditing behavior of the participants as we measured it and we discuss the effects on the level of certainty that can be given by each of the two electorates.

The outcome of our analysis is a negative one: the auditing behavior of people is not sufficient to ensure the correctness of the tally with good probability in either case studied. The same holds true even for simulated data that capture the case of relatively well trained participants while, finally, the security of the ceremony can be shown but under the assumption of essentially ideally behaving human subjects. We note that while our results are stated for Helios, they automatically transfer to various other e-voting systems that, as Helios, rely on client-side encryption to encode the voter's choice.

1 Introduction

A ceremony, introduced by Ellison [Ell07], extends the notion of a security protocol to include "human nodes" in the protocol specification together with regular computer nodes. Human nodes, are compu-

*Research was partly supported by ERC project CODAMODA.

†Research was supported by project FINER, Greek Secretariat of Research and Technology funded under action "ARISTEIA 1."

‡Research was performed while Zhang was a post-doc at National and Kapodistrian University of Athens, Greece.

tationally limited and error-prone; they are able to interact with computer nodes via a user interface (UI) as well as communicate with each other via direct communication lines. In this model, computer nodes can be thought of as stateful and probabilistic interactive Turing machines, while human nodes, even though they are stateful, they are limited in terms of computational power and their behavior can only be considered as a random variable following some arbitrary probability distribution over a set of "admissible behaviors" that are dictated by the UI's they are provided with. Designing and analyzing the security of ceremonies has proven to be valuable for problems that non-trivially rely on human node interaction to ensure their security properties, such as key provisioning and web authentication, see e.g., [Ell07, KTW09, RBNB11, CP12].

In this work, we initiate the study of secure *e-voting ceremonies*. An e-voting ceremony is a protocol between computer and human nodes that aims to assist a subset of the humans (the voters) to cast a ballot for a specified election race. We argue that viewing e-voting as a ceremony captures the security intricacies of the e-voting problem much more effectively than standard protocol based modeling as it was done so far. The reason for this, is that the properties of an election system, most importantly verifiability, rely on human participant behavior in a highly non-trivial manner.

The capability to perform auditing is widely accepted as the most important characteristic for modern e-voting systems. However, even widely deployed[1] systems such as Helios [Adi08] that are touted to be verifiable via auditing still provide only unquantified guarantees of verifiability. The main reason for this is that the correctness of the election result when the election authorities are adversarial is impossible to verify unless the humans that participate in the protocol follow a suitable behavior. This means that the voters, beyond the ballot-casting procedure, are supposed to carry out additional steps that many may find to be counterintuitive, see e.g., [OBV13] for more discussion of this issue. This potentially leads to the defective execution of the appropriate steps that are to be carried out for verifiability to be supported and hence the verifiability of the election may collapse. Recent studies have shown that voters have rather limited participation and interest to perform the verification steps (e.g., [DGK$^+$14] reports about 23 out of a sample of 747 people performed a verifiability check in a deployed end-to-end (E2E) verifiable system). Given that the auditing performed by the voters is critical for the integrity of the election result as a whole, it is imperative to determine the class of distributions of behaviors that are able to detect (significant) misbehavior of the election authorities. Once this class is characterized then one may then try to influence participants to approximate the behavior by training them.

Traditionally, cf. [Cha81, SK95, JCJ02, CMFP$^+$10, Cha04, Nef04], election verifiability was considered at the "individual level" (i.e., a single voter is able to verify her vote intent is properly included in the tally) and the "universal level" (i.e., the election transcript appears to be properly formed). No voter behavioral characteristics were taken into account in the security analysis and the protocols were deemed "end-to-end verifiable" as long as they satisfied merely these two features[2]. The work of [KTV10a, KTV11, KTV12] showed that individual verifiability and universal verifiability, even if combined, can still fail to guarantee that the election tally is correct. To mend the concept of verifiability, a "holistic" notion of global verifiability was introduced. Nevertheless, such global verifiability is unattainable without any assumption on human behavior. Indeed, [KTV12] establishes the verifiability of the Helios system by assuming that voters perform an unbounded number of independent coin flips — an assumption which should be at best considered of theoretical interest, since no voter using the Helios system (or any e-voting system for that matter) should be expected to actually perform ballot-casting via the employment of independent coin flips.

Beyond verifiability, an e-voting system is supposed to also satisfy privacy and other desired properties such as receipt-freeness. These properties interact with verifiability in various important ways:

[1]The web-site of the project reports that more than 100,000 votes have been cast with the system.

[2] A notable departure from this restriction is [ZCC$^+$13], nevertheless no formal security analysis is performed for the verifiability of this system.

First, without privacy it is substantially easier to achieve verifiability (this is due to the fact that verification of the recording of one's vote can be done in relatively straightforward manner assuming a public "bulletin-board" [Ben87]). Second, receipt-freeness combined with verifiability suggests that the receipt obtained by the voter from ballot-casting can be delegated to a third-party without fear of coercion or privacy leakage. Given these reasons, a proper analysis of an e-voting system should also include the analysis of at least these properties.

1.1 Our results

Our results are as follows.

1. We initiate the study of e-voting ceremonies, i.e., e-voting protocols that involve computer and human nodes, and enable the human participant voters to cast privately their ballots and calculate their tally. In an execution of an e-voting ceremony, human nodes follow a certain behavior which is sampled according to some distribution over all possible admissible behaviors. No specific assumptions can be made about how human nodes behave and thus the distribution of each human node is a parameter of the security analysis. It follows that the security properties of e-voting ceremonies are conditional on vectors of probability distributions of human behaviors. Such vectors are specified over sets of suitably defined deterministic finite state machines with output (transducers) that determine all possible ways that each human participant may interact with the UI's of the computer nodes that are available to them.

2. Extending the work of [KTV12, KZZ15], we provide a threat model for (end-to-end) verifiability for e-voting ceremonies. Our threat model has the following characteristics: (i) it provides a holistic approach to argue about end-to-end verifiability by casting the property as an "attack game" played between the adversary and a challenger. (ii) it provides an explicit final goal the adversary wants to achieve by introducing a metric over all possible election outcomes and stating an explicit amount of deviation that the adversary wants to achieve in this metric space. (iii) the adversary is successful provided that the election tally appears to be correct even though it deviates from the true tally according to the stated metric while the number of complaining voters in any failed ballot-casting processes is below a threshold (a ballot-casting process may fail because of adversarial interference). (iv) the resources of the adversary include the complete control of all trustees, election authorities, all voter PC's as well as a subset of the voters themselves. Regarding privacy, we extend the work of [BPW12, KZZ15], by providing a threat model for privacy/receipt-freeness for e-voting ceremonies.

3. We cast Helios as an e-voting ceremony: voters and trustees are the human participants of the protocol that are supposed to handle credentials and receipts as well as generate and validate ciphertexts. During ballot-casting, voters perform the Benaloh challenge process [Ben06] and are free to choose to cast their ballot. Voters may further choose to audit their ballot in the bulletin board if they wish to. Trustees are supposed to execute deterministic steps in order to perform the public-key generation during the setup stage of the election and are able to verify their public-key in the bulletin board if they wish. The set of admissible behaviors for voters include any number of Benaloh challenges followed by casting the ciphertext and choosing whether to audit it in the bulletin board.

4. We analyze the Helios e-voting ceremony with respect to the threat-model for privacy/receipt-freeness and end-to-end verifiability. The behaviors of voters are an explicit component of the

security analysis. Specifically, for end-to-end verifiability, we characterize the space of admissible behaviors that enable the verifiability of the election result and we prove an infeasibility and a feasibility result:

(4.I) it is *infeasible* to detect a large deviation in the published tally of the election even if a high number of voters audit it, if (i) there is some i that the average voter will perform exactly i Benaloh audits with high enough probability compared to the tolerance level of complaints, or (ii) there is a set of indices $\mathcal{J}$ that if the average voter performs $j \in \mathcal{J}$ Benaloh audits, this can be used as a predictor for not auditing the bulletin board; (see Theorem 1 for the precise formulation of the infeasibility result).

(4.II) it is *feasible* to detect a deviation in the tally if a suitable number of voters audit the election, provided that (i) for all i the probability that the adversary performs exactly i Benaloh audits is sufficiently small, and (ii) if the number j of Benaloh audits can be used as a predictor of not auditing the bulletin board, then it holds that the likelihood of j Benaloh audits is sufficiently small; (see Theorem 2 for the precise formulation of the feasibility result).

5. We provide an experimental evaluation from two different sources of human data where people used Helios. We report on the auditing behavior of the participants as we measured it and we discuss the effects on the level of certainty that can be given in each of the two elections.

   The message from our evaluation is a negative one: The behavior profile of people is not such that it can provide sufficient certainty on the correctness of the election result. For instance, as we show from the data collected from the elections of the directors of the International Association for Cryptologic Research (IACR), for elections in the order of hundreds (500) more than 3% of the votes could be overturned with significant probability of no detection (25%), cf. Figure 5. Based on public data on recent election results of the IACR the votes for elected candidates were sufficiently close to candidates that lost in the election and consequently, the results could have been overturned with significant probability without being detected, cf. Table 6. Our results are similarly negative in the second human experiment. Given our negative results for actual human data we turn to simulated results for investigating the case when people are supposedly well trained. Even for a voter behavior distribution with supposedly relatively well trained voters our simulated experiment show that the validity of the election result is sustained with rather low confidence.

We note that even though we focused on Helios in this work, our results (including our threat-model analysis for ceremonies and associated security theorems) immediately apply to a number of other e-voting systems. Such systems (that have been identified as single-pass systems in [BPW12]) include [CFSY96, CGS97, DGS03, KKW06, TPLT13].

1.2 Related work

Modelling verifiability. In [Cha81], Chaum suggested for the first time that anonymous communication can lead to voting systems with individual verifiability.The notion of universal verifiability has been introduced in [SK95], and formally defined in [JCJ02]. Kremer, Ryan and Smyth [KRS10] the verifiability of Helios 2.0 in a symbolic framework framework. Similarly, Smyth *et al.* [SFC], perform an analysis of Helios using a computational framework for verifiability. A formal definition is also provided in [CMFP$^+$10].

End-to-end verifiability in the sense of cast-as-intended, recorded-as-cast, tallied-as-recorded was an outcome of the works in [Cha04] and [Nef04]. The term of E2E verifiability (or more precisely, E2E integrity) also appeared in [Com05]. In [PKRV10], Popoveniuc *et al.* proposed a definition of E2E verifiability via a list of properties. Küsters, Truderung and Vogt [KTV10a] introduced symbolic and computational definitions of verifiability. In [KTV11], showed that individual verifiability and universal verifiability are not sufficient to guarantee the "global" verifiability of an e-voting system and In [KTV12], they introduced a new type of attacks that they name clash attacks, which compromise the integrity of Helios, for variants where the ballots are not linked with the identities of the voters.

Modelling privacy and receipt-freeness. Benaloh and Fischer [CF85] provided a computational definition of privacy while receipt-freeness has been first studied by Benaloh and Tuinstra [BT94]. Chevallier-Mames *et al.* [CMFP+10] introduced definitions for unconditional of privacy and receipt-freeness. Formal definitions for privacy and receipt-freeness have been proposed in the context of applied pi calculus [DKR09] and the universal composability model [Gro04, MN06]. In [KTV11], the level of privacy of an e-voting system is measured w.r.t. to the observation power the adversary has in a protocol run.

In [BCP+11], Bernhard *et al.* proposed a game-based notion of ballot privacy and study the privacy of Helios. Their definition was extended by Bernhard, Pereira and Warinschi [BPW12] by allowing the adversary to statically corrupt election authorities. Both these definitions, although they imply a strong inditinguishability property, do not consider receipt-freeness. We note that our game-based definition captures both privacy and receipt-freeness while restricted to a single EA (and it can easily be extended by including a set of trustees that the adversary may corrupt).

As we have mentioned previously, modelling coercion resistance is out of the scope of this work. We refer the reader to [JCJ02, DKR09, UMQ09, KTV10b, AOZZ15] for formal definitions of coercion resistance in the cryptographic, symbolic and universal composability model. We note that our modeling of receipt-freeness offers only a weak form of coercion resistance (specifically one that the coercer at worst provides the candidate input to the voter prior to ballot-casting but is not forcing the voter to change its program).

1.3 Roadmap

The rest of the paper is organized as follows. In Section 2, we introduce the entities, the syntax and the security framework of an e-voting ceremony. In Section 3, we describe the Helios e-voting ceremony according to our syntax. In Section 4, we analyze the E2E verifiability of Helios ceremony. Namely, we prove (I) an infeasibility and (II) a feasibility result under specific classes of voter behaviors, and we comment on the logical tightness of the two classes. In Section 5, we prove the voter privacy/receipt-freeness of the Helios ceremony. In Section 6, we present evaluations of our results for the E2E verifiability of Helios ceremony. Our evaluations are based on actual human data obtained by elections using Helios as well as simulated data for various sets of parameters. Finally, in the concluding Section 7, we recall the objectives, methodology, analysis and results of this paper and discuss future work.

2 E-Voting Ceremonies

A ceremony [Ell07] is an extension of a network protocol that involves human nodes along side computer nodes. Computer nodes will be modeled in a standard way while we will model humans as probability distributions over a support set of simple finite state machines. We base our framework for ceremonies on the e-voting system modeling from [KZZ15] suitably extending it to our setting.

2.1 The entities of the e-voting ceremony

We consider a security parameter λ that determines the security level of the cryptographic primitives that are being used. Associated with an e-voting ceremony, we consider three additional parameters: the number of voters n, the number of candidates m and the number of trustees k. All three parameters n, m, k are set to be polynomial in λ. We use the notation $\mathcal{P} = \{P_1, ..., P_m\}$ for the set of candidates, $\mathcal{V} = \{V_1, ..., V_n\}$ for the set of voters and $\mathcal{T} = \{T_1, \ldots, T_k\}$ for the set of trustees; the trustees comprise the election committee that guarantees the privacy of the election. In addition, an e-voting ceremony includes (i) an *election authority* (EA) , (ii) a *credential distributor* (CD)[3] and (iii) a publicly accessible *bulletin board* (BB). The entities involved in an e-voting ceremony are shown in Figure 1.

We denote by $\mathcal{U} \subseteq 2^{\mathcal{P}}$ the collection of subsets of candidates that the voters are allowed to choose to vote for (which may also include a "blank" option). We will denote the candidate selection that voter V_ℓ submits to be a subset $\mathcal{U}_\ell \subseteq \mathcal{P}$. Note that in a simple 1-out-of-m voting the set $\mathcal{U}_\ell$ is just a singleton from the subset $\mathcal{P}$.

Next, we define the election function that should be implemented by the e-voting ceremony. For clarity, we only consider election systems that produce the number of votes received by each candidate (this models many standard election procedures). Specifically, let $\mathcal{U}^*$ be the set of vectors of candidate selections of arbitrary length and let f be a function which maps $\mathcal{U}^*$ to the set $\mathbb{Z}_+^m$ so that $f(\mathcal{U}_1, \ldots, \mathcal{U}_n)$ is equal to an m-vector whose j-th location is equal to the number of times P_j was chosen in the candidate selections $\mathcal{U}_1, \ldots, \mathcal{U}_n$.

Modeling human nodes. We model each human node as a collection of simple finite state machines that can communicate with computer nodes (via a user interface) as well as with each other via direct communication. Specifically, we consider a -potentially infinite- collection of *transducers* (i.e. finite state machines with an input and an output tape) that is additionally equipped with a communication tape. We restrict the size of each voter transducer to depend only on the number of candidates m. Note that this has the implication that the voter transducer *cannot be used to perform cryptographic operations* (which require polynomial number of steps in λ). Transducers may interact with computer nodes, (called *supporting devices*) and use them to produce ciphertexts and transmit them to other computer nodes. Transducers corresponding to voter nodes will be denoted as the set $\mathcal{M}^V$ while transducers corresponding to the trustee nodes will be denoted as the set $\mathcal{M}^T$ and transducers corresponding to the credential distributor will be denoted as the set $\mathcal{M}^{\mathrm{CD}}$. We assume that all sets $\mathcal{M}^V, \mathcal{M}^T$ and $\mathcal{M}^{\mathrm{CD}}$ are polynomial time samplable, i.e., one can produce the description of a transducer from the set in polynomial-time and they have an efficient membership test.

2.2 Syntax and Semantics

In order to express the threat model for the e-voting ceremony, we need to formally describe the syntax and semantics of the procedures executed by the ceremony. We think of an e-voting ceremony Π as a quintuple of algorithms and ceremonies denoted by $\langle \mathbf{Setup}, \mathbf{Cast}, \mathbf{Tally}, \mathbf{Result}, \mathbf{Verify} \rangle$ together with the sets of transducers $\mathcal{M}^V, \mathcal{M}^T$ and $\mathcal{M}^{\mathrm{CD}}$ that express the human node operations; these are specified as follows:

•*The ceremony* $\mathbf{Setup}(1^\lambda, \mathcal{P}, \mathcal{V}, \mathcal{U}, \mathcal{T})$ is executed by the EA, a transducer $M^{CD} \in \mathcal{M}^{\mathrm{CD}}$ describing the behavior of CD, the transducers $M_i^T \in \mathcal{M}^T$, $i = 1, \ldots, k$ describing the behavior of the trustees

[3] Note that in practice the CD may be an organization of more than one human nodes executing another ceremony but we do not model this as part of the e-voting ceremony. Here we make the simplifying choice of modeling CD as a single human node (that is able to identify voters using an external identification mechanism operating among humans).

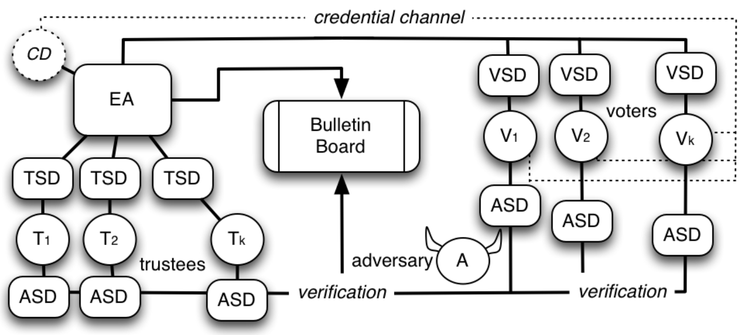


Figure 1: The entities and the channels active in an e-voting ceremony (including the adversary $\mathcal{A}$). The human nodes are the trustees $T_1, \ldots, T_k$, the voters $V_1, \ldots, V_n$ and the credential distributor (CD). The computer nodes are the voting supporting devices (VSDs), the trustee supporting devices (TSDs), the auditing supporting devices (ASDs), the bulletin board (BB) and the election authority (EA). The computer node channels used are shown as solid black lines while the human node channels are shown as dotted lines (there is only one such channel between the credential distributor and the voters). Each human node, voter or trustee, interacts with two computer nodes while the CD interacts with the EA.

$T_1, \ldots T_k$ respectively and their TSDs. The ceremony generates Π's public parameters Pub (which include $\mathcal{P}, \mathcal{V}, \mathcal{U}$) and the voter credentials $s_1, \ldots, s_n$. After the ceremony execution, each TSD has a private state st_i, each trustee T_i obtains a secret $\overline{s}_i$, the EA has a private state $\mathsf{st}_{\mathsf{EA}}$ and the CD obtains the credentials $s_1, \ldots, s_n$. In addition, the EA posts an initial public transcript $\tau = \mathrm{Pub}$ on BB. At the end of the **Setup**, the CD will provide $s_1, \ldots, s_n$ to the voters $V_1, \ldots, V_n$.

•*The ceremony* **Cast** is a five-party ceremony between the EA, BB, VSD, ASD and a transducer $M_{i_\ell} \in \mathcal{M}^V$ that determines the behavior of voter V_ℓ. Voter V_ℓ executes the **Cast** ceremony according to the behavior M_{i_ℓ} as follows: M_{i_ℓ} has input $(s_\ell, \mathcal{U}_\ell)$, where s_ℓ is the voter's credential and $\mathcal{U}_\ell$ represents the candidate selection of V_ℓ. All communication between the voter V_ℓ and EA, BB happens via the VSD of V_ℓ. BB has input τ and EA has input $\mathsf{st}_{\mathsf{EA}}, s_1, \ldots, s_n$. Upon successful termination, M_{i_ℓ}'s output tape contains a receipt α_ℓ returned by the VSD. If the termination is not successful, M_{i_ℓ}'s output tape possibly contains a special symbol '$\mathsf{Complain}$', indicating that voter V_ℓ has decided to complain about the incorrect execution of the election procedure. In any case of termination (successful or not), M_{i_ℓ}'s output tape may contain a special symbol 'Audit', indicating that V_ℓ has taken the decision to use her receipt α_ℓ to perform verification at the end of the election; in this case, the receipt α_ℓ will be provided as input to the ASD of V_ℓ. At the end of the ceremony, EA updates its state and BB updates the public transcript τ as necessary.

•*The ceremony* **Tally** with common input Pub is executed by the EA, the BB, and the trustees $M_i^T \in \mathcal{M}^T$, $i = 1, \ldots, k$ as well as their TSDs. Upon successful termination, BB updates the public transcript τ.

•*The algorithm* **Result**, when given τ as input, outputs the result for the election, or returns $\perp$ in case such result is undefined.

•*The algorithm* **Verify** when given as input τ and a voter receipt α (that corresponds to the voter's output from the **Cast** ceremony execution), returns a value in $\{0, 1\}$.

The correctness of Π is defined as follows:

Definition 1 (Correctness)**.** *The e-voting ceremony* Π *has (perfect)* correctness*, if for any honest execution of* Π *with respect to any CD behavior specified in* $\mathcal{M}^{\mathrm{CD}}$ *that results in a public transcript* τ *where the voters* $V_1, \ldots, V_n$ *cast votes for options* $\mathcal{U}_1, \ldots, \mathcal{U}_n$ *following any of the behaviors specified in* $\mathcal{M}^V$ *and received receipts* $\alpha_1, \ldots, \alpha_n$*, it holds that*

$$\mathbf{Result}(\tau) = f(\mathcal{U}_1, \ldots, \mathcal{U}_n) \textit{ AND } \left(\bigwedge_{\ell=1}^{n} \mathbf{Verify}(\tau, \alpha_\ell) = 1 \right).$$

2.3 Threat model for E2E Verifiability

In order to define the threat model for E2E verifiability we need first to determine the adversarial objective. Intuitively, the objective of the adversary is to manipulate the election result without raising suspicion amongst the participating voters. To express this formally, we have to introduce a suitable notation; given that candidate selections are elements of a set of m choices, we may encode them as m-bit strings, where the bit in the j-th position is 1 if and only if candidate P_j is selected. Further, we may aggregate the election results as the list with the number of votes each candidate has received, thus the output of the $\mathbf{Result}$ algorithm is a vector in $\mathbb{Z}_+^m$. In this case, a result is feasible if and only if the sum of any of its coordinates is no greater than the number of voters.

Vote extractor. Borrowing from [KZZ15], in order to express the threat model for E2E verifiability properly, we will ask for a *vote extractor* algorithm $\mathcal{E}$ (not necessarily efficient, e.g., not running in polynomial-time) that receives as input the election transcript τ and the set of $\mathbf{Cast}$ ceremony receipts $\{\alpha_\ell\}_{\ell \in \mathcal{V}_{\mathsf{succ}}}$, where by $\mathcal{V}_{\mathsf{succ}}$, we denote the set of honest voters that voted successfully. Given such input, $\mathcal{E}$ will attempt to compute $n - |\mathcal{V}_{\mathsf{succ}}|$ vectors $\langle \mathcal{U}_\ell \rangle_{V_\ell \in \mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}}$ in $\{0,1\}^m$ which correspond to all the voters outside of $\mathcal{V}_{\mathsf{succ}}$ and can be either a candidate selection, if the voter has voted adversarially or a zero vector, if the voter has not voted successfully. In case $\mathcal{E}$ is incapable of presenting such selection, the symbol $\perp$ will be returned instead. The purpose of the algorithm $\mathcal{E}$ is to express the requirement that the election transcript τ that is posted by the EA in the BB at the end of the procedure contains (in potentially encoded form) a set of well-formed actual votes. Using this notion of extractor, we are capable to express the "actual" result encoded in an election transcript despite the fact that the adversary controls some voters. Note when the extractor $\mathcal{E}$ fails it means that τ is meaningless as an election transcript and thus unverifiable.

Election result deviation. Next, we want to define a measure of *deviation* from the actual election result, as such deviation is the objective of the adversary in an E2E verifiability attack. This will complete the requirements for expressing the adversarial objective in the E2E attack game. To achieve this, it is natural to equip the space of results with a *metric*. We use the metric derived by the 1-norm, $\|\cdot\|_1$ scaled to half, i.e.,

$$\begin{array}{lll} \mathrm{d}_1: & \mathbb{Z}_+^m \times \mathbb{Z}_+^m & \longrightarrow \mathbb{R} \\ & (w, w') & \longmapsto \frac{1}{2} \cdot \|w - w'\|_1 = \frac{1}{2} \cdot \sum_{i=1}^{n} |w_i - w'_i|, \end{array}$$

where w_i, w'_i is the i-th coordinate of w, w' respectively.

Let $R \in \mathbb{Z}_+^m$ be the election results that correspond to the true voter intent of n voters, and $R' \in \mathbb{Z}_+^m$ be the published election results. Denote by $\max(\mathcal{U})$, the maximum cardinality of an element in $\mathcal{U}$. Then, two encodings of candidate selections are within $\max(\mathcal{U})$ distance, so intuitively, if the adversary wants to present u' as the result of the election, it may do that by manipulating the votes of at least $\mathrm{d}_1(R, R')/\max(\mathcal{U})$ voters. This means that e.g., in simple 1-out-of-m voting, moving i votes from one candidate to another translates to a distance $\mathrm{d}_1(R, R')$ of exactly i.

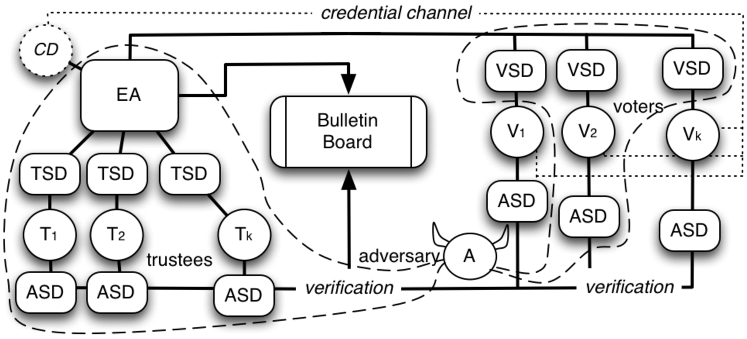


Figure 2: The adversarial entities during an E2E verifiability attack.

The E2E verifiability game. Let $\mathcal{D} = \langle \mathbf{D}_1, \ldots, \mathbf{D}_n, \mathbf{D}_1^T, \ldots, \mathbf{D}_k^T, \mathbf{D}^{\mathrm{CD}} \rangle$ be a vector of distributions that consists of the distributions $\mathbf{D}_1, \ldots, \mathbf{D}_n$ over the collection of voter transducers $\mathcal{M}^V$, the distributions $\mathbf{D}_1^T, \ldots, \mathbf{D}_k^T$ over the collection of trustee transducers $\mathcal{M}^T$ and the distribution $\mathbf{D}^{\mathrm{CD}}$ over the collection of CD transducers $\mathcal{M}^{\mathrm{CD}}$. We define the *E2E Verifiability game*, denoted by $G_{\text{E2E-Ver}}^{\mathcal{A},\mathcal{E},\mathcal{D},d,\theta,\phi}$, between the adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ using a voter extractor $\mathcal{E}$ with parameters d, θ, ϕ (defined below). The game takes as input the security parameter λ, the number of voters n, the number of candidates m, and the number of trustees k. The entities that are adversarially controlled in the game are presented in Figure 2.

The attack game is parameterized by (i) the deviation amount, d, (according to the metric $\mathrm{d}_1(\cdot,\cdot)$) that the adversary wants to achieve, (ii) the number of honest voters, θ, that terminate the $\mathbf{Cast}$ ceremony successfully and (iii) the number of honest voters, ϕ, that complain in case of unsuccessful termination during the $\mathbf{Cast}$ ceremony. The adversary starts by selecting the voter, candidate and trustee identities for given parameters n, m, k. It also determines the allowed ways to vote, $\mathcal{U}$. The adversary now fully controls the trustees, the EA as well as all the VSD's while the CD remains honest during the setup stage. The adversary manages the $\mathbf{Cast}$ ceremony executions where it assumes the role of both the EA and the VSD. For each voter, the adversary may choose to corrupt it or to allow the challenger to play on its behalf. Note that the challenger retains the control of the ASD for honest voters and samples for each honest voter a transducer from the corresponding distribution. If a voter V_ℓ is uncorrupted, the adversary provides the candidate selection that V_ℓ should use in the $\mathbf{Cast}$ ceremony; the challenger samples a transducer $M_{i_\ell} \leftarrow \mathbf{D}_\ell$ from voter transducer distribution $\mathbf{D}_\ell$ and then executes the $\mathbf{Cast}$ ceremony according to M_{i_ℓ}'s description to vote the given candidate selection and decide whether to audit the election result at the end. The adversary finally posts the election transcript to the BB. The adversary will win the game provided that there is a subset of cardinality at least θ of honest voters that terminate the ballot-casting successfully and at most ϕ complaining honest voters, but the deviation of the tally is bigger than d or the extractor fails to produce the candidate selection of the dishonest voters. The attack game is specified in detail in Figure 3.

Definition 2 (E2E-Verifiability)**.** *Let* $0 < \epsilon < 1$ *and* $n, m, k, d, \theta, \phi \in \mathbb{N}$ *with* $\theta, \phi \leq n$. *The e-voting ceremony* Π *w.r.t. the election function* f *achieves* E2E verifiability *with error* ϵ*, transducer distribution vector* $\mathcal{D}$*, a number of at least* θ *honest successful voters, at most* ϕ *honest complaining voters and tally deviation at most* d *if there exists a (not necessarily polynomial-time) vote extractor* $\mathcal{E}$ *such that for every PPT adversary* $\mathcal{A}$*:*

$$\Pr[G_{\text{E2E-Ver}}^{\mathcal{A},\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda, n, m, k) = 1] < \epsilon.$$

E2E Verifiability Game $G_{\text{E2E-Ver}}^{\mathcal{A},\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda, n, m, k)$

1. $\mathcal{A}$ chooses the sets of candidates $\mathcal{P} = \{P_1, ..., P_m\}$, voters $\mathcal{V} = \{V_1, ..., V_n\}$, and trustees $\mathcal{T} = \{T_1, ..., T_k\}$, and the set of allowed candidate selections $\mathcal{U}$.
2. $\mathcal{C}$ performs the **Setup** ceremony on input $(1^\lambda, \mathcal{P}, \mathcal{V}, \mathcal{U}, \mathcal{T})$ with the adversary playing the role of EA and all trustees and their associated TSDs while $\mathcal{C}$ plays the role of CD (Refer to Fig. 2 for an overview of the corrupted nodes) by following the transducer $M^{\text{CD}} \leftarrow \mathbf{D}^{\text{CD}}$. In this way $\mathcal{C}$ obtains the voter credentials $\{s_\ell\}_{\ell\in[n]}$. If the CD refuses to distribute the credentials to the voters, then the game terminates.
3. The adversary $\mathcal{A}$ and the challenger $\mathcal{C}$ engage in an interaction where $\mathcal{A}$ schedules the **Cast** ceremonies of all voters. For each voter V_ℓ, $\mathcal{A}$ can either completely control the voter or allow $\mathcal{C}$ operate on their behalf, in which case $\mathcal{A}$ provides a candidate selection $\mathcal{U}_\ell$ to $\mathcal{C}$. In order to perform the ceremony, $\mathcal{C}$ samples a transducer $M_{i_\ell} \leftarrow \mathbf{D}_\ell$ and engages with the adversary $\mathcal{A}$ in the **Cast** ceremony so that $\mathcal{A}$ plays the role of VSD and EA and $\mathcal{C}$ plays the role of V_ℓ according to transducer M_{i_ℓ} on input $(s_\ell, \mathcal{U}_\ell)$ and its associated ASD. Provided the ceremony terminates successfully, $\mathcal{C}$ obtains the receipt α_ℓ produced by M_{i_ℓ}, on behalf of V_ℓ.
4. Finally, $\mathcal{A}$ posts the election transcript τ to the BB.

We define the following subsets of honest voters (i.e., those controlled by $\mathcal{C}$):

- $\mathcal{V}_{\mathsf{succ}}$ is the set of honest voters that terminated successfully.
- $\mathcal{V}_{\mathsf{comp}}$ is the set of honest voters s.t. the special symbol '$\mathsf{Complain}$' is written on the output tape of the corresponding transducer.
- $\mathcal{V}_{\mathsf{audit}}$ is the set of honest voters s.t. the special symbol 'Audit' is written on the output tape of the corresponding transducer.

The game returns a bit which is 1 if and only if the following conditions hold true:

(i). $|\mathcal{V}_{\mathsf{succ}}| \geq \theta$.

(ii). $|\mathcal{V}_{\mathsf{comp}}| \leq \phi$.

(iii). $\forall \ell \in [n]$: if $V_\ell \in \mathcal{V}_{\mathsf{audit}}$ then $\mathbf{Verify}(\tau, \alpha_\ell) = 1$.

and either one of the following two conditions:

(iv-a). if $\perp \neq \langle \mathcal{U}_\ell \rangle_{V_\ell \in \mathcal{V}\setminus\mathcal{V}_{\mathsf{succ}}} \leftarrow \mathcal{E}(\tau, \{\alpha_\ell\}_{V_\ell\in\mathcal{V}_{\mathsf{succ}}})$, then $\mathrm{d}_1(\mathbf{Result}(\tau), f(\langle \mathcal{U}_1, \dots, \mathcal{U}_n \rangle)) \geq d$.

(iv-b). $\perp \leftarrow \mathcal{E}(\tau, \{\alpha_\ell\}_{V_\ell\in\mathcal{V}_{\mathsf{succ}}})$.

Figure 3: The E2E Verifiability Game between the challenger $\mathcal{C}$ and the adversary $\mathcal{A}$ using the vote extractor $\mathcal{E}$ and w.r.t. the vector of transducer distributions $\mathcal{D} = \langle \mathbf{D}_1, \dots, \mathbf{D}_n, \mathbf{D}_1^T, \dots, \mathbf{D}_k^T, \mathbf{D}^{\text{CD}} \rangle$.

Remark 1 (Universal voter distribution). We have introduced the collection of transducers $\mathcal{M}^V, \mathcal{M}^T, \mathcal{M}^{\text{CD}}$ to model all possible admissible behaviors that voters, trustees and credential distributors respectively might follow to successfully complete the e-voting ceremony. Note that in the security modeling of the e-voting ceremony, each voter V_ℓ is associated with a distribution $\mathbf{D}_\ell$ over $\mathcal{M}^V$, which captures its voter profile. For instance, the voter V_1 may behave as transducer M_1 with 50% probability, M_2 with 30% probability, and M_3 with 20% probability. In some e-voting systems, the voters can be uniquely identified during the **Cast** ceremonies, e.g. the voter's real ID is used. Hence, the adversary is able to identify each voter V_ℓ and learn its profile expressed by $\mathbf{D}_\ell$. Then, the adversary may choose the best attack strategy depending on $\mathbf{D}_\ell$. Nevertheless, in case the credentials are randomly and anonymously assigned to the voters by the CD, the adversary will not be able to profile voters given his view in the ballot-casting ceremony (recall that in the E2E game the CD remains honest). Therefore, it is possible to unify the distributions to a *universal voter* distribution, denoted as $\mathbf{D}$, which reflects the profile of the "average voter." Specifically, in this case, we will have $\mathbf{D}_1 = \cdots = \mathbf{D}_n = \mathbf{D}$.

2.4 Threat model for Voter Privacy (including Receipt-Freeness)

The threat model of privacy concerns the actions that may be taken by the adversary to figure out the choices of the honest voters. We specify the goal of the adversary in a very general way. In particular, for an attack against privacy to succeed, we ask that there is an election result, for which the adversary is capable of distinguishing how people vote while it has access to (i) the actual receipts that the voters obtain after ballot-casting as well as (ii) a set of ceremony views that are consistent with all the honest voters' views in the $\mathbf{Cast}$ ceremony instances they participate.

Observe that any system that is secure against such a threat scenario possesses also "receipt-freeness", in the sense that voters cannot prove how they voted by showing the receipt they obtain from the $\mathbf{Cast}$ ceremony or even presenting their view in the $\mathbf{Cast}$ ceremony. Given that in the threat model we allow the adversary to observe the view of the voter in the $\mathbf{Cast}$ ceremony, we need to allow the voter to be able to lie about her view in the ceremony (otherwise an attack could be trivially mounted). We stress that the simulated view of the voter in the $\mathbf{Cast}$ ceremony does not contain the view of the internals of the VSD. This means that, with respect to privacy, the adversary may not look into the internals of the VSD for the honest voters. The above is consistent, for instance, with the scenario that the voter can give to the VSD her candidate choice to be encoded. While the adversary will be allowed to observe a simulated view of the voter during the $\mathbf{Cast}$ ceremony, it will be denied access to the internals of the VSD during the $\mathbf{Cast}$ execution. This increases the opportunities where the voter can lie about how she executes the $\mathbf{Cast}$ ceremony.

The Voter Privacy/Receipt-freeness Game. Following the same logic as in the E2E Verifiability game, we specify a vector of transducer distributions over the collection of voter transducers $\mathcal{M}^V$, trustee transducers $\mathcal{M}^T$ and CD transducers $\mathcal{M}^{\mathrm{CD}}$ denoted by $\mathcal{D} = \langle \mathbf{D}_1, \ldots, \mathbf{D}_n, \mathbf{D}_1^T, \ldots, \mathbf{D}_k^T, \mathbf{D}^{\mathrm{CD}} \rangle$. We then express the threat model as a *Voter Privacy game*, denoted by $G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}$, that is played between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, that takes as input the security parameter λ, the number of voters n, the number of candidates m, and the number of trustees k and returns 1 or 0 depending on whether the adversary wins. An important feature of the voter privacy game is the existence of an *efficient simulator* $\mathcal{S}$ that provides a simulated view of the voter in the $\mathbf{Cast}$ ceremony. Note that the simulator is not responsible to provide the view of the voter's supporting device (VSD). Intuitively, this simulator captures the way the voter can lie about her choice in the $\mathbf{Cast}$ ceremony in case she is coerced to present her view after she completes the ballot-casting procedure. The parties controlled by the adversary during a privacy attack are presented in Figure 4.

The attack game is parameterized by t. The adversary starts by selecting the voter, candidate and trustee identities for given parameters n, m, k. It also determines the allowed ways to vote and selects a single trustee to remain honest together with its TSD and ASD. The challenger subsequently flips a coin b (that will change its behavior during the course of the game) and will perform the $\mathbf{Setup}$ ceremony with the adversary playing the role of the CD and of all the trustees and their associated TSDs and ASDs except one trustee that will remain honest. The honest trustee behavior will be determined by a transducer that is selected at random by the challenger from $\mathcal{M}^T$ according to the corresponding distribution. Subsequently, the adversary will schedule all $\mathbf{Cast}$ ceremonies selecting which voters it prefers to corrupt and which ones it prefers to allow to vote honestly.

The adversary is allowed to corrupt at most t voters and their VSDs. In addition, $\mathcal{A}$ is allowed to corrupt the ASDs of all voters. The voters that remain uncorrupted are operated by the challenger and they are given two candidate selections to vote. For each uncorrupted voter V_ℓ, the challenger first

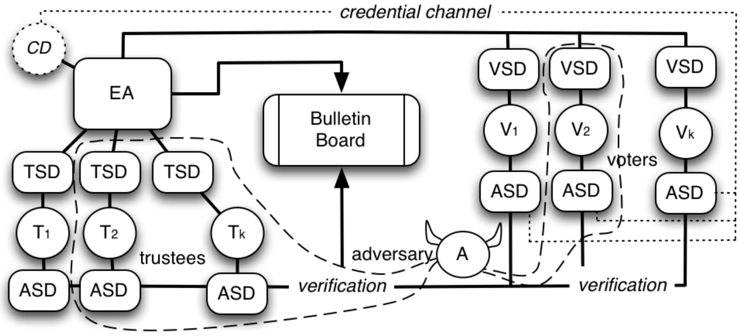


Figure 4: The adversarial entities during an attack against voter privacy/receipt-freeness.

samples a transducer $M_{i_\ell} \leftarrow \mathbf{D}_\ell$ and then executes the **Cast** ceremony according to M_{i_ℓ}'s description to vote one of its two candidate selections based on b. The adversary will also receive the receipt that is obtained by each voter as well as either (i) the actual view (if $b = 0$) or (ii) a *simulated* view, generated by $\mathcal{S}$ (if $b = 1$), of each voter during the **Cast** ceremony (this addresses the receipt-freeness aspect of the attack game). Upon completion of ballot-casting, the adversary and the challenger will executethe **Tally** ceremony and subsequently, the adversary will attempt to guess b. The attack is successful provided that the election result is the same with respect to the two alternatives provided for each honest voter by the adversary and the adversary manages to guess the challenger's bit b correctly. The game is presented in detail in Figure 5.

Definition 3 (Voter Privacy/Receipt-Freeness)**.** *Let $n, m, k, t \in \mathbb{N}$ with $t \leq n$. The e-voting ceremony Π w.r.t. the election function f achieves* voter privacy/receipt-freeness *for at most t corrupted voters and for transducer distribution vector $\mathcal{D}$, if there is an efficient simulator $\mathcal{S}$ such that for any PPT adversary $\mathcal{A}$:*

$$\left|\Pr[G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}(1^\lambda, n, m, k) = 1] - 1/2\right| = \mathsf{negl}(\lambda).$$

Remark 2*.* Our game-based voter privacy/receipt-freeness definition is close in spirit to witness indistinguishability of interactive proof systems. A potentially stronger privacy requirement would be a simulation-based formulation (akin to zero-knowledge in interactive proof systems) e.g., as the one suggested for ballot privacy in [BPW12]. Here we opt to extend the privacy model of [KZZ15].

Remark 3 (Corruption of the credential distributor)*.* In our framework, we assumed that the CD can be malicious in the voter privacy/receipt freeness game while it is kept honest for E2E verifiiability. This choice is made for consistency with the level of security that Helios [Adi08] as well as most *client-side encryption* e-voting systems can provide regarding credential distribution (e.g. [CGS97, JCJ05]). Namely, since the vote is encrypted in the voter's VSD, knowing the credential of the voter alone does not suffice for breaking her privacy. On the other side, for E2E verifibiality it is important that an honest authority verifies the uniqueness of the credentials, otherwise the election is susceptible to "clash attacks" [KTV12]. If one wishes to study the security of *votecode-based* e-voting systems (e.g. [Cha01, CEC+08, KZZ15]), then they would have to take the opposite approach. In such systems, the credentials contain encodings of the candidates that are personal for each voter, therefore the CD has to be honest for voter privacy/receipt freeness. On the other hand, these systems have mechanisms during the **Cast** ceremony, that inherently guarantee resistance against clash attacks, hence corrupting the CD does not affect their E2E verifiability.

Voter Privacy/Receipt-freeness Game $G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}(1^\lambda, n, m, k)$

1. $\mathcal{A}$ on input $1^\lambda, n, m, k$, chooses a list of candidates $\mathcal{P} = \{P_1, ..., P_m\}$, a set of voters $\mathcal{V} = \{V_1, ..., V_n\}$, a set of trustees $\mathcal{T} = \{T_1, ..., T_k\}$ a trustee $T_w \in \mathcal{T}$ and the set of allowed candidate selections $\mathcal{U}$. It provides $\mathcal{C}$ with the sets $\mathcal{P}, \mathcal{V}, \mathcal{U}$ as well as the trustee identity T_w.

2. $\mathcal{C}$ flips a coin $b \in \{0, 1\}$ and performs the **Setup** ceremony on input $(1^\lambda, \mathcal{P}, \mathcal{V}, \mathcal{U}, \mathcal{T})$ with the adversary playing the role of the CD and all trustees except T_w, while $\mathcal{C}$ plays the role of EA and T_w as well as T_w's TSD. The roles of T_w is played by $\mathcal{C}$ following the transducers $M^{T_w} \leftarrow \mathbf{D}^{T_w}$ (Refer to Fig. 4 for an overview of the corrupted nodes).

3. The adversary $\mathcal{A}$ and the challenger $\mathcal{C}$ engage in an interaction where $\mathcal{A}$ schedules the **Cast** ceremonies of all voters which may run concurrently. $\mathcal{A}$ also controls the ASDs of all voters. At the onset of each voter ceremony, $\mathcal{A}$ chooses whether voter V_ℓ, $\ell = 1, \ldots, n$ and its associated VSD is corrupted or not.

   - If V_ℓ and its associated VSD are corrupted, then no specific action is taken by the challenger, as the execution is internal to adversary.
   - If V_ℓ and its associated VSD are not corrupted, then $\mathcal{A}$ provides $\mathcal{C}$ with two candidate selections $\langle \mathcal{U}_\ell^0, \mathcal{U}_\ell^1 \rangle$. The challenger samples $M_{i_\ell} \leftarrow \mathbf{D}_\ell$ and sets V_ℓ's input to $(s_\ell, \mathcal{U}_\ell^b)$, where s_ℓ is the credential provided by the adversarially controlled CD. Then, $\mathcal{C}$ and $\mathcal{A}$ engage in the **Cast** ceremony with $\mathcal{C}$ controlling V_ℓ (that behaves according to M_{i_ℓ}), V_ℓ's VSD, and the EA, while the adversary $\mathcal{A}$ observes the network interaction. When the **Cast** ceremony terminates, the challenger $\mathcal{C}$ provides to $\mathcal{A}$: (i) the receipt α_ℓ that V_ℓ obtains from the ceremony, and (ii) <u>if $b = 0$</u>, the current view of the internal state of the voter V_ℓ that the challenger obtains from the **Cast** execution, or <u>if $b = 1$</u>, a simulated view of the internal state of V_ℓ produced by $\mathcal{S}(view_{\mathcal{C}})$, where $view_{\mathcal{C}}$ is the current view of the challenger.

4. $\mathcal{C}$ performs the **Tally** ceremony playing the role of EA, T_w and its associated TSD following M_w^T while $\mathcal{A}$ plays the role of all other trustees.

5. Finally, $\mathcal{A}$ terminates returning a bit b^*.

Let $\mathcal{V}_{\mathsf{succ}}$ be the set of voters that terminate the voting ceremony successfully without being corrupted. The game returns a bit which is 1 if and only if the following hold true:

(i). $b = b^*$ (i.e., the adversary guesses b correctly).

(ii). $|\mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}| \leq t$ (i.e., number of corrupted voters is bounded by t).

(iii). $f(\langle \mathcal{U}_\ell^0 \rangle_{V_\ell \in \mathcal{V}_{\mathsf{succ}}}) = f(\langle \mathcal{U}_\ell^1 \rangle_{V_\ell \in \mathcal{V}_{\mathsf{succ}}})$ (i.e., the election result w.r.t. the set of voters $\tilde{\mathcal{V}}$ does not leak b).

Figure 5: The Voter-privacy/Receipt-freeness attack game between the challenger $\mathcal{C}$ and the adversary $\mathcal{A}$ using the simulator $\mathcal{S}$ and w.r.t. the vector of transducer distributions $\mathcal{D} = \langle \mathbf{D}_1, \ldots, \mathbf{D}_n, \mathbf{D}_1^T, \ldots, \mathbf{D}_k^T, \mathbf{D}^{\mathrm{CD}} \rangle$.

3 Syntax of Helios Ceremony

In this section, we present a formal description of Helios ceremony according to the syntax provided in Section 2.2. For simplicity, we consider the case of *1-out-of-m elections*, where the set of allowed selections $\mathcal{U}$ is the collection of singletons, $\{\{P_1\}, \ldots, \{P_m\}\}$, from the set of candidates $\mathcal{P}$. Our syntax does not reflect the current implemented version of Helios, as it adapts necessary minimum modifications to make Helios secure. For instance, we ensure that each voter is given a *unique identifier* to prevent Helios from the clash attacks introduced in [KTV12]. In addition, we consider a hash function $H(\cdot)$ that all parties have oracle access to, used for committing to election information and ballot generation, as well as the *Fiat-Shamir transformations* in the non-interactive zero-knowledge (NIZK) proofs that the system requires. As we state below, in the generation of the NIZK proofs for ballot correctness, the unique identifier is included in the hash to prevent replaying attacks presented in [CS10]. Moreover, we need to use strong Fiat-Shamir transformations, where the statement of the NIZK should also be included in the hash. As shown in [BPW12], strong Fiat-Shamir based NIZKs are *simulation sound extractable*, while weak Fiat-Shamir based NIZKs make the Helios vulnerable.

The Helios's transducers. We define the description of the collections of transducers $\mathcal{M}^V, \mathcal{M}^T, \mathcal{M}^{\mathrm{CD}}$ for all the admissible behaviors of voters, trustees and credential distributors respectively.

The set of admissible voter transducers is denoted by $\mathcal{M}^V := \{M_{i,c,a}\}_{i\in[0,q]}^{c,a\in\{0,1\}}$, where $q \in \mathbb{N}$; The transducer $M_{i,c,a}$ audits the ballot created by the VSD exactly i times (using its ASD) and then submits the $(i+1)$-th ballot created by the VSD; Upon successful termination, it outputs a receipt α obtained from the VSD; If the termination is not successful and $c = 1$, $M_{i,c,a}$ outputs a special symbol '$\mathsf{Complain}$' to complain about its failed engagement in the $\mathbf{Cast}$ ceremony. In any case of termination, when $a = 1$, $M_{i,c,a}$ also outputs a special symbol 'Audit' and sends the receipt α to the ASD. In order to guarantee termination, we limit the maximum number of ballot audits by threshold q.

The admissible trustee transducer M^T is simple and unique (so that $\mathcal{M}^T = \{M^T\}$). At high level, M^T will utilize the TSD to generate a partial public/secret key pair in the $\mathbf{Setup}$ ceremony. We assume it performs this perfectly so there is no variability in its operation.

The credential distributor is required to check the validity of the credentials $s_1, \ldots, s_n$ generated by the potentially malicious EA before distributing them. In Helios, we define the credential $s_i := (\mathrm{ID}_i, t_i)$, where ID_i is a unique voter identity and t_i is an authentication token. The credential distributor first checks forall $i, j \in [n]$: if $i \neq j$ then $\mathrm{ID}_i \neq \mathrm{ID}_j$, and halts if the verification fails. Upon success, it randomly sends each voter V_ℓ a credential though some human channels. Hence, we define the set of CD transducers as $\mathcal{M}^{\mathrm{CD}} := \left\{M_j^{\mathrm{CD}}\right\}_{j\in S_n}$, where S_n stands for all possible permutations $[n] \mapsto [n]$.

We define the Helios ceremony quintuple $\langle\mathbf{Setup}, \mathbf{Cast}, \mathbf{Tally}, \mathbf{Result}, \mathbf{Verify}\rangle$ as follows:

• $\mathbf{Setup}(1^\lambda, \mathcal{P}, \mathcal{V}, \mathcal{U}, \mathcal{T})$: sequentially, each trustee transducer M_i^T, $i = 1, \ldots, k$ sends signal to its TSD. The TSD generates a pair of ElGamal partial keys $(\mathsf{pk}_i, \mathsf{sk}_i)$ and sends pk_i together with a (strong Fiat-Shamir) NIZK proof of knowledge of sk_i to EA. In addition, the TSD returns a trustee secret $\bar{s}_i := (H(\mathsf{pk}_i), \mathsf{sk}_i)$ to M_i^T. If there is a proof that EA does not verify, then EA aborts the protocol. Next, EA computes the election public key $\mathsf{pk} = \prod_{i=1}^k \mathsf{pk}_i$. The public parameters, Pub, which include the election information denoted by Info, pk and the partial public keys as well as their NIZK proofs of knowledge are posted in the BB by the EA. Then M_i^T, $i = 1, \ldots, k$ sends $H(\mathsf{pk}_i)$ to its ASD, and the ASD will fetch Pub from the BB to verify if there exists a partial public key pk_* such that its hash matches $H(\mathsf{pk}_i)$. The EA then generates the voter credentials $s_1, \ldots, s_n$, where $s_i := (\mathrm{ID}_i, t_i)$, and forwards them to the CD transducer M^{CD}. The CD transducer M^{CD} checks the uniqueness of each ID_i and then sends them to the voter transducer M_{i_ℓ,c_ℓ,a_ℓ} for $\ell \in [n]$.

• The $\mathbf{Cast}$ ceremony is described by the following. For each voter V_ℓ, the corresponding transducer M_{i_ℓ,c_ℓ,a_ℓ} has a pre-defined number of i_ℓ ballot auditing steps, where $i_\ell \in [0, q]$.

The input of M_{i_ℓ,c_ℓ,a_ℓ} is $(s_\ell, \mathcal{U}_\ell)$. For $u \in [i_\ell]$, the following steps are executed:

M_{i_ℓ,c_ℓ,a_ℓ} sends $(\mathrm{ID}_\ell, \mathcal{U}_\ell)$ to its VSD. Let P_{j_ℓ} be the candidate selection of V_ℓ, i.e. $\mathcal{U}_\ell = \{P_{j_\ell}\}$. For $j = 1, \ldots, m$, VSD creates a ciphertext, $C_{\ell,j}$, that is a lifted ElGamal encryption under pk of 1, if $j = j_\ell$ (the selected candidate position), or 0 otherwise. In addition, it attaches a NIZK proof $\pi_{\ell,j}$ showing that $C_{\ell,j}$ is an encryption of 1 or 0. Finally, an overall NIZK proof π_ℓ is generated, showing that exactly one of these ciphertexts is an encryption of 1. These proofs are strong Fiat-Shamir transformations of *disjunctive Chaum-Pedersen* proofs. To generate the proofs, the unique identifier ID_ℓ is included in the hash. The ballot generated is $\psi_{\ell,u} = \langle\psi_{\ell,u}^0, \psi_{\ell,u}^1\rangle$, where $\psi_{\ell,u}^0 = \langle(C_{\ell,1}, \pi_{\ell,1}), \ldots, (C_{\ell,m}, \pi_{\ell,m}), \pi_\ell)\rangle$ and $\psi_{\ell,u}^1 = H(\psi_{\ell,u}^0)$. The VSD responds to M_{i_ℓ,c_ℓ,a_ℓ} with the ballot $\psi_{\ell,u}$. Then, M_{i_ℓ,c_ℓ,a_ℓ} sends a *Benaloh* *audit request* to the VSD. In turn, VSD returns the randomness $r_{\ell,u}$ that was used to create the ballot $\psi_{\ell,u}$. The M_{i_ℓ,c_ℓ,a_ℓ} sends $(\mathrm{ID}_\ell, \psi_{\ell,u}, r_{\ell,u})$ to its ASD, which will audit the validity of the ballot. If the verification fails, M_{i_ℓ,c_ℓ,a_ℓ} halts. If the latter happens and $c_\ell = 1$, M_{i_ℓ,c_ℓ,a_ℓ} outputs a special symbol '$\mathsf{Complain}$', otherwise it returns no output.

After the i_ℓ-th successfully Benaloh audit, M_{i_ℓ,c_ℓ,a_ℓ} invokes the VSD to produce a new ballot ψ_ℓ as before; however, upon receiving ψ_ℓ, M_{i_ℓ,c_ℓ,a_ℓ} now sends s_ℓ to the VSD to indicate it to submit the ballot to the EA. The M_{i_ℓ,c_ℓ,a_ℓ} then outputs $\alpha_\ell := (\mathrm{ID}_\ell, \psi_\ell^1)$. If $a_\ell = 1$, M_{i_ℓ,c_ℓ,a_ℓ} also outputs a special symbol 'Audit' which indicates that it will send α_ℓ to its ASD which will audit the BB afterwards, as specified in the $\mathbf{Verify}$ algorithm below.

When EA receives a cast vote ψ_ℓ, it checks that it is a well-formed ballot by verifying the NIZK proofs. If the check fails, then it aborts the protocol. After voting ends, EA updates its state with the pairs $\{(\psi_\ell, \mathrm{ID}_\ell)\}_{V_\ell \in \mathcal{V}_{\mathsf{succ}}}$ of cast votes and the associated identifiers, where $\mathcal{V}_{\mathsf{succ}}$ is the set of voters that voted successfully.

• In the $\mathbf{Tally}$ ceremony, EA sends $\{\psi_\ell\}_{V_\ell \in \mathcal{V}_{\mathsf{succ}}}$ to all trustee transducers M_i^T's TSD, $i = 1, \dots, k$. For every trustee T_i, $i = 1, \dots, k$, the corresponding transducer M_i^T is the previously defined transducer M^T. Next, the TSD of each M_i^T, performs the following computation: it constructs the product ciphertext $\mathbf{C}_j = \prod_{V_\ell \in \mathcal{V}_{\mathsf{succ}}} C_{\ell,j}$ for $j = 1, \dots, m$. By the additive homomorphic property of (lifted) ElGamal, each $\mathbf{C}_j$ is a valid encryption of the number of votes that the candidate P_j received. Then, the TSD uses sk_i to produce the partial decryption of all C_j, denoted by x_j^i, and sends it to the EA along with NIZK proofs of correct partial decryption. The latter are Fiat-Shamir transformations of *Chaum-Pedersen* proofs of discrete log equality. If there is a proof that EA does not verify, then it aborts the protocol. After all trustees finish their computation, EA updates τ with $\{(x_1^i, \dots, x_m^i)\}_{i \in [k]}$ and the NIZK proofs.

• For each candidate P_j, the $\mathbf{Result}$ algorithm computes the number of votes, x_j, that P_j has received using the partial decryptions $x_j^1, \dots, x_j^k$. The output of the algorithm is the vector $\langle x_1, \dots, x_m \rangle$.

• The algorithm $\mathbf{Verify}(\tau, \alpha_\ell)$ outputs 1 if the following conditions hold:

1. The structure of τ and all election information is correct (using Info).
2. There exists a ballot in τ, indexed by ID_ℓ, that contains the hash value ψ_ℓ^1.
3. The NIZK proofs for the correctness of all ballots in τ verify.
4. The NIZK proofs for the correctness of all trustees' partial decryptions verify.
5. For $j = 1, \dots, m$, x_j is a decryption of $\mathbf{C}'_j$, where $\mathbf{C}'_j$ is the homomorphic ciphertext created by multiplying the respective ciphertexts in the ballots published on the BB (in an honest execution $\mathbf{C}'_j$ should be equal to $\mathbf{C}_j$).

4 E2E Verifiability of an e-voting ceremony

In a Helios e-voting ceremony, an auditor can check the correct construction of the ballots and the valid decryption of the homomorphic tally by verifying the NIZK proofs. In our analysis, it is sufficient to require that all NIZK proofs have negligible soundness error ϵ in the *random oracle* (RO) model. Note that in Section 3, we explicitly modify Helios to associate ballots with the voters' identities, otherwise a clash attack [KTV12] would break verifiability. For simplicity in presentation, we assume that the identifiers are created by the adversary, i.e. the set $\{\mathrm{ID}_\ell\}_{\ell \in [n]}$ matches the set of voters $\mathcal{V}$.

Throughout our analysis, we assume the honesty of the CD and thus the distribution of the credentials is considered to be an arbitrary permutation over $[n]$. Since there is only one admissible trustee transducer M^T, the distribution of trustee transducers $\mathbf{D}^T$ is set as:

$$\Pr_{\mathbf{D}^T}[M] = \begin{cases} 1 & \text{if } M = M^T \\ 0 & \text{if } M \neq M^T \end{cases} \tag{1}$$

Moreover, in the **Cast** ceremony, the ballots and receipts are produced before the voters show their credentials to the system. Since the CD is honest, the adversary is oblivious the the maps between the credentials to the voter transducers. Moreoever, the credentials are only required when the voters want to submit their ballots. Hence, according to the discussion in Remark 1, we will consider only a universal voter transducer distribution $\mathbf{D}$ in the case study of Helios. Namely, $\mathbf{D}_1 = \cdots = \mathbf{D}_n = \mathbf{D}$.

In Section 4.1 we describe the types of non-trivial attacks on the verifiabiilty of Helios, which may also be launched in other e-voting systems that, like Helios, use client-side encryption and allow BB auditing. In Section 4.2, we describe an adversarial strategy against the verifiabiilty of Helios and prove that is effective against a specified class of *assailable* voter transducer distributions. In Section 4.3, we prove a feasibility of E2E verifiability of Helios, for another class of *resistant* voter transducer definitions. Finally, in Section 4.4 we comment on the logical tightness of the two classes.

4.1 Attacks on the verifiability of Helios

As we mentioned in the introduction of this section, we have modified Helios to prevent the system from clash attacks [KTV12]. For simplicity, we exclude all the trivial attacks that the adversary may follow, i.e. the ones that will be detected with certainty (e.g. malformed or unreadable voting interface and public information). Therefore, the meaningful types of attack that an adversary may launch are the following:

– **Collision attack:** the adversary computes two votes which hash to the same value. The collision resistance of the hash function $H(\cdot)$, prevents from these attacks except from some negligible probability ϵ'[4].
– **Invalid vote attack:** the adversary creates a vote for some invalid plaintext, i.e. a vector that does not encode a candidate selection (e.g., multiple votes for some specific candidate). This attack can be prevented by the soundness of the NIZK proofs, except from the negligible soundness error ϵ (verification is done via the voter's ASD).
– **VSD attack:** the adversary creates a vote which is valid, but corresponds to different selection than the one that the voter intended. A Benaloh audit at the **Cast** ceremony step can detect such an attack with certainty, as the randomness provided by the VSD perfectly binds the plaintext with the audited ElGamal ciphertext.
– **BB attack:** the adversary deletes/inserts an honest vote from/to the BB, or replaces it with some other vote of its choice, after voting has ended. Assuming no hash collisions, any such modification will be detected if the voter chooses to audit the BB via her ASD.
– **Invalid tally decryption attack:** the adversary provides a decryption which is not the plaintext that the homomorphic tally vector encrypts. The NIZK proofs of correct decryption prevent this attack, except for a negligible soundness error ϵ.

Remark 4 (Completeness of the attack list). It can be easily shown that the above list exhausts all possible attack strategies against Helios in our threat model. Namely, in an environment with no clash, collision and invalid encryption attacks, the set of votes is in the correct (yet unknown) one-to-one

[4]This requires that $H(\cdot)$ has resistance to second pre-image attacks.

correspondence with the set of voters, and all votes reflect a valid candidate selection of the unique corresponding voter. As a result, a suitably designed vote extractor will decrypt (in super-polynomial time) and output the actual votes from the non-honest-and-succesful voters, up to permutation. Consequently, if no honest vote has been modified during and after voting, and the homomorphic tally of the votes is correctly computed, then the perfect binding of the plaintexts and ciphertexts of ElGamal implies that the decryption of the tally is the *intended election result*.

4.2 Attacking the verifiability of an e-voting ceremony

As explained in the previous section, any attempt of collision, invalid vote and invalid tally decryption attacks has negligible probability of success for the adversary due to the collision resistance of the hash function and the soundness of the ZK proofs. Therefore, in a setting where no clash attacks are possible, the adversary's chances to break verifiability rely on combinations of VSD and BB attacks. The probablity of these attacks being detected depends on the voter transducer distribution $\mathbf{D}$ which depicts their auditing behavior during and after voting. In the following theorem, we prove that the verifiability of Helios is susceptible to VSD or/and BB attacks, when the voters sample from a class of assailable voter transducer distributions.

Theorem 1. *Assume an election run of Helios with n voters, m candidates and k trustees. Let $q, d, \theta, \phi \in \mathbb{N}$, where $0 < \theta, \phi \leq n$ and q is the maximum number of Benaloh audits. Let $\mathbf{D}$ be a (universal) voter transducer distribution s.t. for some $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2 \in [0, 1)$ at least one of the two following conditions holds:*

(i). There is an $i^* \in \{0, \ldots, q\}$ that determines "vulnerable VSD auditing behavior"*. Namely, (i.a) the probability that a voter executes at least i^* Benaloh audits is $1 - \kappa_1$ AND (i.b) the probability that a voter, given that she has executed at least i^* Benaloh audits, will cast her vote after exactly i^* Benaloh audits is $1 - \kappa_2$ AND (i.c) the probability that a voter, given that she will execute exactly i^* Benaloh audits, will not complain in case of unsuccessful audit is κ_3.*

(ii). There is a subset $\mathcal{J} \subseteq \{0, \ldots, q\}$ that determines "vulnerable BB auditing behavior"*. Namely, (ii.a) the probability that a voter executes j Benaloh audits for some $j \in \mathcal{J}$ is $1 - \mu_1$ AND (ii.b) for every $j \in \mathcal{J}$, the probability that a voter, given she has executed j Benaloh audits, will not audit the BB is at least $1 - \mu_2$.*

Let $\mathcal{D} = \langle \mathbf{D}, \ldots, \mathbf{D}, \mathbf{D}_1^T, \ldots, \mathbf{D}_k^T, \mathbf{D}^{CD} \rangle$ be a transducer distribution vector where $\mathbf{D}_i^T = \mathbf{D}^T$, $i = 1, \ldots, k$, is the fixed trustee transducer distribution in Eq. (1) and $\mathbf{D}^{CD}$ is an arbitrary CD trsansducer distribution. Then, there is a PPT adversary $\mathcal{A}$ that wins the E2E verifiability game $G_{\mathrm{E2E-Ver}}^{\mathcal{A},\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda, n, m, k)$ for any vote extractor $\mathcal{E}$, any $\delta \in [0, 1)$ as follows:

- *under condition (i), provided the parameters d, θ, ϕ satisfy:*

$$\begin{aligned} d &\leq (1-\delta)^2(1-\kappa_2)(1-\kappa_1)n \\ \theta &\leq n - (1+\delta)(\kappa_2 + \delta - \delta\kappa_2)(1-\kappa_1)n \\ \phi &\geq (1+\delta)^2\kappa_3(\kappa_2 + \delta - \delta\kappa_2)(1-\kappa_1)n \end{aligned}$$

  with probability of success at least $\boxed{1 - 5e^{-\kappa_3\beta_2\beta_1\frac{\delta^2}{3}}}$

  where $\beta_1 = (1-\delta)(1-\kappa_1)n$ and $\beta_2 = (\kappa_2 - \delta + \delta\kappa_2)(1-\kappa_2)$.

- *under condition (ii), provided the parameter d satisfies $d \leq (1-\delta)(1-\mu_1)n$*

  with probability of success at least $\boxed{(1-e^{-(1-\mu_1)n\frac{\delta^2}{2}})(1-\mu_2)^d}$.

Proof. We observe that when an adversary makes no voter corruptions, then the set $\mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}$ contains only honest voters that did not complete the $\mathbf{Cast}$ ceremony successfully. Therefore, the election result w.r.t. $\mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}$ is zero, so in our analysis we can fix the trivial vote extractor $\mathcal{E}$ that outputs the zero vector of length $|\mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}|$. By definition, if the adversary breaks the E2E verifiability game for $\mathcal{E}$, then it does so for any other vote extractor.

We denote by $E_{i,c,a}$ the event that the honest voter engages in the $\mathbf{Cast}$ ceremony by running the transducer $M_{i,c,a}$. We study the following two cases:

Case 1. *Condition (i) holds [Breaking verifiabiliy via VSD attacks].* We describe a PPT adversary $\mathcal{A}_1$ against verifiabiilty as follows: $\mathcal{A}_1$ corrupts no voters and observes the number of Benaloh audits that each voter performs. If the voter has executed i^* Benaloh audits, then $\mathcal{A}_1$ performs a VSD attack on the i^*+1-th ballot that the voter requests.

By condition (i.a), the probablity that the voter will perform at least i^* Benaloh audits is $\Pr_{\mathbf{D}}[\neg\big(\bigvee_{\substack{0\leq i<i^* \\ c,a\in\{0,1\}}} E_{i,c,a}\big)] = 1-\kappa_1$. Let T be the number of VSD attacks that $\mathcal{A}_1$ executes. It is easy to see that T follows the binomial distribution $B(n, 1-\kappa_1)$. Therefore, by the Chernoff bounds we have that for any $\delta \in [0,1)$,

$$\begin{aligned} &\Pr_{\mathbf{D}}[(1-\delta)(1-\kappa_1)n < T < (1+\delta)(1-\kappa_1)n] \geq \\ &\geq 1 - e^{-(1-\kappa_1)n\delta^2/2} - e^{-(1-\kappa_1)n\frac{\delta^2}{\min\{2+\delta,3\}}} \geq 1 - 2e^{-(1-\kappa_1)n\delta^2/3}. \end{aligned} \tag{2}$$

Let X_T be the number of successful VSD attacks out of all T attempts. Observe that each successful single VSD attack adds 1 to the total tally deviation (the ballot encrypts a candidate vector that is different from the voter's intented selection). Hence, $\mathcal{A}_1$ achieves tally deviation exactly X_T. By condition (i.b), the probablity that a voter, given that it has executed at least i^* Benaloh audits, will execute exactly i^* Benaloh audits is $\Pr_{\mathbf{D}}[\bigvee_{c,a\in\{0,1\}} E_{i^*,c,a} | \neg\big(\bigvee_{\substack{0\leq i<i^* \\ c,a\in\{0,1\}}} E_{i,c,a}\big)] = 1-\kappa_2$. By definition, X_T follows the binomial distribution $B(T, 1-\kappa_2)$. Thus, by the Chernoff bounds we have that for any $\delta \in [0,1)$,

$$\begin{aligned} &\Pr_{\mathbf{D}}[(1-\delta)(1-\kappa_2)T < X_T < (1+\delta)(1-\kappa_2)T] \geq \\ &\geq 1 - e^{-(1-\kappa_2)T\delta^2/2} - e^{-(1-\kappa_2)T\frac{\delta^2}{\min\{2+\delta,3\}}} \geq 1 - 2e^{-(1-\kappa_2)T\delta^2/3}. \end{aligned} \tag{3}$$

According to the description of $\mathcal{A}_1$, the number of honest voters that will not complete the $\mathbf{Cast}$ ceremony successfully is $T - X_T \geq 0$. Therefore, the number of successful honest voters is $|\mathcal{V}_{\mathsf{succ}}| = n - (T - X_T)$. In addition, by condition (i.c), the number of complaining voters $|\mathcal{V}_{\mathsf{comp}}|$ follows the binomial distribution $B(T - X_T, \kappa_3)$. Hence, by the Chernoff bounds and for any $\delta \in [0,1)$,

$$\Pr_{\mathbf{D}}[|\mathcal{V}_{\mathsf{comp}}| < (1+\delta)\kappa_3(T - X_T)] \geq 1 - e^{-\kappa_3(T-X_T)\delta^2/3}. \tag{4}$$

By description, $\mathcal{A}_1$ will definitely win the game $G_{\mathrm{E2E-Ver}}^{\mathcal{A}_1,\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda, n, m, k)$ when

$$\big(X_T \geq d\big) \wedge \big(n - (T - X_T) \geq \theta\big) \wedge \big(|\mathcal{V}_{\mathsf{comp}}| \leq \phi\big).$$

Based on the above observation, we provide a lower bound on the probability that $\mathcal{A}_1$ wins the E2E verifiabiilty game $G_{\mathrm{E2E-Ver}}^{\mathcal{A}_1,\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda, n, m, k)$ when the parameters d, θ, ϕ satisfy the following constraints:

$$d \leq \big(1-\delta\big)^2(1-\kappa_1)(1-\kappa_2)n \tag{5a}$$

$$\theta \leq n - (1+\delta)(\kappa_2 + \delta - \delta\kappa_2)(1-\kappa_1)n \tag{5b}$$

$$\phi \geq (1+\delta)^2\kappa_3(\kappa_2 + \delta - \delta\kappa_2)(1-\kappa_1)n \tag{5c}$$

By Eq. (2),(3) and (4), we have that

$$\begin{aligned}
&\Pr_{\mathbf{D}}[G_{\mathrm{E2E-Ver}}^{\mathcal{A}_1,\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda,n,m,k)=1] \geq \Pr_{\mathbf{D}}\Big[\big(X_T \geq (1-\delta)^2(1-\kappa_2)(1-\kappa_1)n\big)\wedge\\
&\quad\wedge\big(|\mathcal{V}_{\mathsf{comp}}| \leq (1+\delta)^2\kappa_3(\kappa_2+\delta-\delta\kappa_2)((1-\kappa_1)n\big)\wedge\\
&\quad\wedge\big(T-X_T\big) \leq (\kappa_2+\delta-\delta\kappa_2)(1+\delta)(1-\kappa_1)n\Big] \geq\\
&\geq (1-2e^{-(1-\kappa_1)n\delta^2/3})\cdot(1-2e^{-(1-\kappa_2)[(1-\delta)(1-\kappa_1)n]\delta^2/3})\cdot\\
&\quad\cdot(1-e^{-\kappa_3([1-(1+\delta)(1-\kappa_2)]\cdot[(1-\delta)(1-\kappa_1)n])\frac{\delta^2}{\min\{2+\delta,3\}}}) \geq\\
&\geq 1-5e^{-\kappa_3(\kappa_2-\delta+\delta\kappa_2)(1-\kappa_2)(1-\delta)(1-\kappa_1)n\delta^2/3} = 1-5e^{-\kappa_3\beta_2\beta_1\frac{\delta^2}{3}},
\end{aligned} \tag{6}$$

where $\beta_1=(1-\delta)(1-\kappa_1)n$ and $\beta_2=(\kappa_2-\delta+\delta\kappa_2)(1-\kappa_2)$.

Case 2. *Condition (ii) holds [Breaking verifiabiliy via BB attacks].* We describe a PPT adversary $\mathcal{A}_2$ against verifiabiilty as follows: $\mathcal{A}_2$ makes no corruptions and keeps record of the voters that perform j Benaloh audits for some $j \in \mathcal{J}$. Let $\mathcal{V}_{\mathcal{J}}$ be the set of those voters. After all **Cast** ceremonies have been completed, every voter has terminated successfully, i.e. $\mathcal{V}_{\mathsf{succ}}=\mathcal{V}$ and $\mathcal{V}_{\mathsf{comp}}=\emptyset$. In order to achieve tally deviation d, $\mathcal{A}_2$ performs a BB attack on the votes of an arbitrary subset of d voters in $\mathcal{V}_{\mathcal{J}}$. As in the previous case, each single BB attack adds 1 to the total tally deviation, so $|\mathcal{V}_{\mathcal{J}}| \geq d$ must hold. By condition (ii.a), the probability $\Pr_{\mathbf{D}}[\bigvee_{\substack{j\in\mathcal{J}\\ c,a\in\{0,1\}}} E_{j,c,a})]$ that a voter is in $\mathcal{V}_{\mathcal{J}}$ is $1-\mu_1$. By definition, $|\mathcal{V}_{\mathcal{J}}|$ follows the binomial distribution $B(n,1-\mu_1)$. Thus, by the Chernoff bound and for any $\delta\in[0,1)$,

$$\Pr_{\mathbf{D}}[|\mathcal{V}_{\mathcal{J}}| > (1-\delta)(1-\mu_1)n] \geq 1-e^{(1-\mu_1)n\delta^2/2}. \tag{7}$$

However, $\mathcal{A}_2$ will be successful iff all d voters in the selected subset of $\mathcal{V}_j$ do not audit the BB. By condition (ii.b) and the independency of the voter transducers' sampling, this happens with probability at least $(1-\mu_2)^d$. Therefore by Eq. (7), we have that for $d \leq (1-\delta)(1-\mu_1)n$ and any θ,ϕit holds that

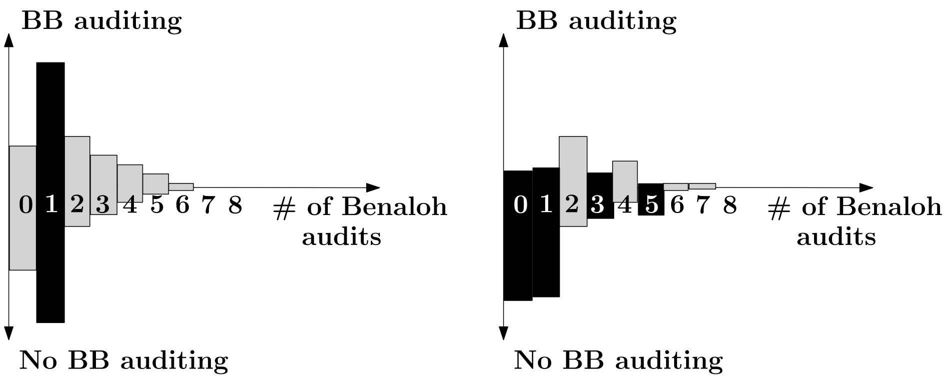
$$\begin{aligned}
&\Pr[G_{\mathrm{E2E-Ver}}^{\mathcal{A}_2,\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda,n,m,k)=1] =\\
&=\Pr[\big(G_{\mathrm{E2E-Ver}}^{\mathcal{A}_2,\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda,n,m,k)=1\big)\wedge(|\mathcal{V}_{\mathcal{J}}| \geq (1-\delta)(1-\mu_1)n)] \geq\\
&\geq (1-e^{-(1-\mu_1)n\delta^2/2})(1-\mu_2)^d.
\end{aligned} \tag{8}$$

By the lower bounds provided in Eq. (6),(8) and by combining the constraints (5a),(5b),(5c) and $d \leq (1-\delta)(1-\mu_1)n$, we get the complete proof of the theorem. ■

Illustrating Theorem 1. To provide intuition, illustrate two representatives from the class of assailable voter transducer distributions that correspond to conditions (i) and (ii) of Theorem 1 in Figures 6(a) and 6(b) respectively, where the length of the bars is proportional to the probability of the corresponding event.

4.3 Proof of verifiability for an e-voting ceremony

In this section, we prove the E2E verifiability of Helios e-voting ceremony in the random oracle model, when the voter transducer distribution satisfies two conditions. As we will explain at lengh in the next section, these conditions are logically complementary to the conditions in the statement of Theorem 1, as long as the complaining behavior of the voters is balanced (i.e. the voters have 1/2 probability of complaining in case of unsuccessful termination).



(a) A voter transducer distribution with vulnerable VSD auditing behavior ($i^* = 1$).

(b) A voter transducer distribution with vulnerable BB auditing behavior ($\mathcal{J} = \{0, 1, 3, 5\}$).

Figure 6: Assailable voter transducer distributions for Helios e-voting ceremony.

Theorem 2. *Assume an election run of Helios with n voters, m candidates and k trustees. Assume that the hash function $H(\cdot)$ considered in Section 3 is a random oracle. Let $q, d, \theta, \phi \in \mathbb{N}$, where $0 < \theta, \phi \leq n$ and q is the maximum number of Benaloh audits. Let $\mathbf{D}$ be a (universal) transducer distribution and some $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2 \in [0, 1)$ s.t. the two following conditions hold:*

- *(i) There is an $i^* \in \{0, \ldots, q+1\}$ that guarantees* "resistance against VSD attacks"*. Namely, (i.a) the probability that a voter executes at least i^* Benaloh audits is κ_1 and (i.b) for every $i \in \{0, \ldots, q\}$, if $i < i^*$, then the probability that a voter, given that she will execute at least i Benaloh audits, will cast her vote after exactly i Benaloh audits, is no more than κ_2 AND the probability that a voter, given that she will execute exactly i Benaloh audits, will complain in case of unsuccessful audit is at least $1 - \kappa_3$.*
- *(ii) There is a subset $\mathcal{J} \subseteq \{0, \ldots, q\}$ that guarantees* "resistance against BB attacks"*. Namely, (ii.a) the probability that a voter executes j Benaloh audits for some $j \in \mathcal{J}$ is $1 - \mu_1$ AND (ii.b) for every $j \in \mathcal{J}$, the probability that a voter, given she has executed j Benaloh audits, will audit the BB is at least $1 - \mu_2$.*

Let $\mathcal{D} = \langle \mathbf{D}, \ldots, \mathbf{D}, \mathbf{D}_1^T, \ldots, \mathbf{D}_k^T, \mathbf{D}^{CD} \rangle$ be a transducer distribution vector where $\mathbf{D}_i^T = \mathbf{D}^T$, $i = 1, \ldots, k$, is the fixed trustee transducer distribution in Eq. (1) and $\mathbf{D}^{CD}$ is an arbitrary CD trsansducer distribution. Then, for any $\delta \in [0, 1)$ and parameters

$$
\begin{aligned}
&d \geq 2(1+\delta)\max\{\kappa_1, \mu_1\}n \\
&\theta \geq n - \left(\frac{1}{(1+\delta)\kappa_2} - 1\right)\left(d/2 - (1+\delta)\kappa_1 n\right) \\
&\phi \leq (1-\delta)(1-\kappa_3)\left(\frac{1}{(1+\delta)\kappa_2} - 1\right)\left(d/2 - (1+\delta)\kappa_1 n\right),
\end{aligned}
$$

the Helios e-voting ceremony achieves E2E verifiability for $\mathcal{D}$, a number of θ honest successful voters, a number of ϕ honest complaining voters and tally deviation d with error

$$\boxed{5\min\{\mu_2^{-1}, e\}^{-(1-\kappa_3)\min\{\kappa_1 n, \gamma_2\gamma_1\}\frac{\delta^2}{3}} + (\gamma_3)^\theta + \mathsf{negl}(\lambda)}$$

where $\gamma_1 = d/2 - (1+\delta)\max\{\kappa_1, \mu_1\}n$, $\gamma_2 = \min\{\frac{1}{(1+\delta)\kappa_2} - 1, \kappa_2\}$ and $\gamma_3 = \mu_1 + \mu_2 - \mu_1\mu_2$.

Proof. Construction of the vote extractor for Helios. The vote extractor $\mathcal{E}$ for Helios receives as input τ and the set of receipts (list of IDs paired with hashes) $\{\alpha_\ell\}_{\mathcal{V}_{\mathsf{succ}}}$. Then, $\mathcal{E}$ executes the following steps:

The vote extractor $\mathcal{E}(\tau, \{\alpha_\ell\}_{\mathcal{V}_{\mathsf{succ}}})$:

1. If the result is not meaningful (i.e., $\mathbf{Result}(\tau) = \perp$), then $\mathcal{E}$ outputs $\perp$. Otherwise, $\mathcal{E}$ arbitrarily arranges the voters in $\mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}$ as $\langle V_\ell^{\mathcal{E}} \rangle_{n-|\mathcal{V}_{\mathsf{succ}}|}$.
2. For every $\ell \in [n - |\mathcal{V}_{\mathsf{succ}}|]$:
   (a) $\mathcal{E}$ reads the vote list in τ. It locates the first vote, denoted by $\psi_\ell^{\mathcal{E}}$, which neither includes a hash appearing in $\{\alpha_\ell\}_{V_\ell \in \mathcal{V}_{\mathsf{succ}}}$, nor is associated with some voter in $\mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}$, and associates this vote with $V_\ell^{\mathcal{E}}$. If no such vote exists, then $\mathcal{E}$ sets $\mathcal{U}_\ell^{\mathcal{E}} = \emptyset$ (encoded as the zero vector).
   (b) $\mathcal{E}$ decrypts the ciphertexts in $\psi_\ell^{\mathcal{E}}$ (in superpolynomial time). If the decrypted messages form a vector in $\{0,1\}^m$ that has 1 in a single position, j_ℓ , then it sets $\mathcal{U}_\ell^{\mathcal{E}} = \{P_{j_\ell}\}$. Otherwise, it outputs $\perp$.
3. Finally, $\mathcal{E}$ outputs $\langle \mathcal{U}_\ell^{\mathcal{E}} \rangle_{V_\ell^{\mathcal{E}} \in \mathcal{V} \setminus \mathcal{V}_{\mathsf{succ}}}$.

Assume a PPT adversary $\mathcal{A}$ that wins the game $G_{\mathrm{E2E-Ver}}^{\mathcal{A},\mathcal{E},\mathbf{D},d,\theta,\phi}(1^\lambda, n, m, k)$, for the above vote extractor $\mathcal{E}$. We denote by i_ℓ the number of Benaloh audits that the honest voter V_ℓ executes. We denote by $E_{i,c,a}$ the event that the voter engages in the $\mathbf{Cast}$ ceremony by running the transducer $M_{i,c,a}$.

Let A be the event that at least one honest voter will audit the BB after voting, i.e. $\mathcal{V}_{\mathsf{audit}} \neq \emptyset$. By condition *(ii)*, the probability that $V_\ell \notin \mathcal{V}_{\mathsf{audit}}$ is bounded by

$$\begin{aligned}
&\Pr_{\mathbf{D}}[V_\ell \notin \mathcal{V}_{\mathsf{audit}}] = \Pr_{\mathbf{D}}[E_{i_\ell,0,0} \vee E_{i_\ell,1,0}] = \\
&= \Pr_{\mathbf{D}}[(E_{i_\ell,0,0} \vee E_{i_\ell,1,0}) \wedge i_\ell \in \mathcal{J}] + \Pr_{\mathbf{D}}[(E_{i_\ell,0,0} \vee E_{i_\ell,1,0}) \wedge i_\ell \notin \mathcal{J}] \leq \\
&\leq \Pr_{\mathbf{D}}[i_\ell \in \mathcal{J}] + (1 - \Pr_{\mathbf{D}}[i_\ell \in \mathcal{J}]) \cdot \Pr_{\mathbf{D}}[E_{i_\ell,0,0} \vee E_{i_\ell,1,0} \mid i_\ell \notin \mathcal{J}] \leq \\
&\leq \mu_1 + (1 - \mu_1)\mu_2 = \mu_1 + \mu_2 - \mu_1\mu_2 .
\end{aligned} \tag{9}$$

Therefore, by Eq. (9), the independence of the transducers' sampling and the fact that there are at least θ honest (and successful) voters, we have that

$$\Pr[\neg A] = \Pr_{\mathbf{D}}[\bigwedge_{V_\ell \in \mathcal{V}_{\mathsf{succ}}} (V_\ell \notin \mathcal{V}_{\mathsf{audit}})] \leq (\mu_1 + \mu_2 - \mu_1\mu_2)^\theta . \tag{10}$$

Let F be the event that $\mathcal{A}$ has performed at least one invalid vote or tally decryption attack. Namely, one of the homomorphic tally ciphertexts $\mathbf{C}_j$, for $j \in [m]$, does not decrypt as x_j, or a ballot of a voter $V_\ell \in \mathcal{V}$ does not correspond to an encryption of a vector in $\{0,1\}^m$ that has 1 in a single position. Assuming that $H(\cdot)$ is a RO, all the NIZK proofs are sound except from a negligible error ϵ. If $\mathcal{V}_{\mathsf{audit}} \neq \emptyset$, there is at least one honest voter who verifies the ZK proofs. Hence, it holds that

$$\Pr[\big(G_{\mathrm{E2E-Ver}}^{\mathcal{A},\mathcal{E},d,\theta,\phi}(1^\lambda, n, m, k) = 1\big) \wedge F \mid A] \leq \epsilon = \mathsf{negl}(\lambda) . \tag{11}$$

Suppose that F does not occur. In this case, $\mathcal{E}$ outputs a vector of selections that is a permutation of the adversarial votes and some zero vectors, thus it homomorphically sums to the actual adversarial result. Therefore, $\mathcal{A}$ deviates from the intended result $f(\langle \mathcal{U}_1, \ldots, \mathcal{U}_n \rangle)$ only because it (i) alters some votes of the voters in $\mathcal{V}_{\mathsf{succ}}$ during voting or (ii) replaces, deletes or inserts some of the votes of the (successful or unsuccessful) honest voters in τ (BB). By Remark 4, $\mathcal{A}$ achieves this by performing combinations of collision, VSD and BB attacks. As mentioned in Section 4.1, the probability of a successful collision attack ($\mathcal{A}$ provides V_ℓ with a receipt α_ℓ that has the same hash value as a another ballot of $\mathcal{A}$'s choice) is no more than a negligible function ϵ'.

We denote by X the set of the honest voters whose votes have been altered during voting (VSD attack) and by Y the set of honest voters whose votes have been replaced/deleted/inserted in the BB, both determined by $\mathcal{A}$'s adaptive strategy. Each of these attacks adds 1 to the total deviation, so the deviation that $\mathcal{A}$ achieves is $|X \cup Y| = |X \setminus Y| + |Y| \geq d$. Since the parameter d is at least $2(1+\delta)\max\{\kappa_1, \mu_1\}n$, we have that (1) $|X \setminus Y| \geq d/2 \geq (1+\delta)\max\{\kappa_1, \mu_1\}n$ or (2) $|Y| \geq d/2 \geq (1+\delta)\max\{\kappa_1, \mu_1\}n$ must hold. We examine the probability that all the attacks on the voters in $X \setminus Y$ and Y are successful for both cases.

(1) $|X \setminus Y| \geq d/2 \geq (1+\delta)\max\{\kappa_1, \mu_1\}n$ **holds.** Let T the set of voters that $\mathcal{A}$ attempted a VSD attack. We partition T, X into the following sets:

$$T_< = \{V_\ell \in T | i_\ell < i^*\} \quad \text{and} \quad T_\geq = \{V_\ell \in T | i_\ell \geq i^*\}$$
$$X_< = \{V_\ell \in X | i_\ell < i^*\} \quad \text{and} \quad X_\geq = \{V_\ell \in X | i_\ell \geq i^*\},$$

where i^* is defined in condition *(i)* of the theorem's statement. Clearly, $X_< \subseteq T_<$ and $X_\geq \subseteq T_\geq$. By condition *(i.a)*, $|T_\geq|$ is a random variable that follows the binomial distribution $Bin(n, \kappa_1)$. By condition *(i.b)*, for an arbitrary value z, the probability $\Pr_{\mathbf{D}}[|X_<| \geq z]$ is no more than $\Pr[|\tilde{X}_<| \geq z]$, where $|\tilde{X}_<|$ is a random variable that follows the binomial distribution $Bin(|T_<|, \kappa_2)$.

By the syntax of Helios ceremony, the voters can complain only when they are under under VSD atack, so it holds that $\mathcal{V}_{\mathsf{comp}} \subseteq T$. Thus, we can partition the set of complaining voters $\mathcal{V}_{\mathsf{comp}}$ into the two sets $\mathcal{V}^<_{\mathsf{comp}} = \mathcal{V}_{\mathsf{comp}} \cap T_<$ and $\mathcal{V}^\geq_{\mathsf{comp}} = \mathcal{V}_{\mathsf{comp}} \cap T_\geq$. By condition *(i.b)*, for an arbitrary value z, the probability $\Pr_{\mathbf{D}}[|\mathcal{V}^<_{\mathsf{comp}}| \leq z]$ is no more than $\Pr[|\tilde{\mathcal{V}}^<_{\mathsf{comp}}| \leq z]$, where $|\tilde{\mathcal{V}}^<_{\mathsf{comp}}|$ follows the binomial distribution $Bin(|T_<| - |X_<|, 1 - \kappa_3)$. By the Chernoff bounds, we have that for any $\delta \in [0, 1)$, the following hold:

1. If $|T_\geq| < (1+\delta)\kappa_1 n$, then $|X_\geq| \leq |T_\geq| < (1+\delta)\kappa_1 n$, so $\mathcal{A}$ wins only if

$$|X_<| = |X| - |X_\geq| > |X \setminus Y| - (1+\delta)\kappa_1 n \geq d/2 - (1+\delta)\kappa_1 n.$$

2. If $[|X_<| < (1+\delta)\kappa_2 |T_<|$, then $|T_<| - |X_<| > \Big(\dfrac{1}{(1+\delta)\kappa_2} - 1\Big)|X_<|$, so

$$\begin{aligned}
&\Pr_{\mathbf{D}}\big[(|\mathcal{V}^<_{\mathsf{comp}}| \leq (1-\delta)(1-\kappa_3)\big(\frac{1}{(1+\delta)\kappa_2} - 1\big)|X_<|) \mid |X_<| < (1+\delta)\kappa_2|T_<|\big)\big] \leq \\
&\leq \Pr_{\mathbf{D}}[(|\mathcal{V}^<_{\mathsf{comp}}| \leq (1-\delta)(1-\kappa_3)(|T_<| - |X_<|) \mid |X_<| < (1+\delta)\kappa_2|T_<|)] \leq \\
&\leq e^{-(1-\kappa_3)\big(\frac{1}{(1+\delta)\kappa_2} - 1\big)|X_<|)\delta^2/2}.
\end{aligned}$$

Based on the above observations, we have that for any $\delta \in [0,1)$,

$$
\begin{aligned}
&\Pr[\mathcal{A} \text{ successful in } X \setminus Y] = \Pr_{\mathbf{D}}\big[(\mathcal{A} \text{ successful in } X \setminus Y) \wedge \big(|T_{\geq}| \geq (1+\delta)\kappa_1 n\big)\big] + \\
&\quad + \Pr_{\mathbf{D}}\big[(\mathcal{A} \text{ successful in } X \setminus Y) \wedge \big(|T_{\geq}| < (1+\delta)\kappa_1 n\big)\big] \leq \\
&\leq \Pr_{\mathbf{D}}[|T_{\geq}| \geq (1+\delta)\kappa_1 n] + \Pr_{\mathbf{D}}\Big[\big(|X_{<}| \geq d/2 - (1+\delta)\kappa_1 n\big) \wedge \\
&\quad \wedge \big(|\mathcal{V}_{\mathsf{comp}}| \leq (1-\delta)(1-\kappa_3)\big(\frac{1}{(1+\delta)\kappa_2} - 1\big)\big(d/2 - (1+\delta)\kappa_1 n\big)\big) \mid \\
&\qquad\qquad \mid \big(|T_{\geq}| < (1+\delta)\kappa_1 n\big)\Big] \leq \\
&\leq e^{-\kappa_1 n \delta^2/3} + e^{-\kappa_2 (d/2-(1+\delta)\kappa_1 n)\delta^2/3} + e^{-(1-\kappa_3)\left(\frac{1}{(1+\delta)\kappa_2}-1\right)(d/2-(1+\delta)\kappa_1 n)\delta^2/2} \leq \\
&\leq 3e^{-(1-\kappa_3)\min\left\{\kappa_1 n, \min\{\frac{1}{(1+\delta)\kappa_2}-1, \kappa_2\}\left(d/2-(1+\delta)\kappa_1 n\right)\right\}\delta^2/3}.
\end{aligned}
\tag{12}
$$

(2) $|Y| \geq d/2 \geq (1+\delta)\max\{\kappa_1, \mu_1\}n$ holds. A replacement/deletion/insertion attack may be successful because (a) $\mathcal{A}$ has computed an adversarial ballot with the same hash values ψ_ℓ (collision attack) or (b) V_ℓ is not in $\mathcal{V}_{\mathsf{audit}}$. Given the subset $\mathcal{J}$ in condition *(ii)* of the stament, we partition Y into the subsets: $Y_a = \{V_\ell \in Y | i_\ell \in \mathcal{J}\}$ and $Y_b = \{V_\ell \in Y | i_\ell \notin \mathcal{J}\}$. By condition *(ii.a)*, $|Y_b|$ follows the binomial distribution $Bin(n, \mu_1)$. Moreover, by condition *(ii.b)*, the probability of a successful BB attack against any voter in Y_a is upper bounded by $\mu_2 + \epsilon'$ (the voter does not audit the BB or $\mathcal{A}$ finds a collision). Finally, when $|Y_b| < (1+\delta)\mu_1 n$, then $|Y_a| = |Y| - |Y_b| \geq d/2 - (1+\delta)\mu_1 n$. Thus, by the Chernoff bounds and for any $\delta \in [0,1)$,

$$
\begin{aligned}
&\Pr[\mathcal{A} \text{ successful in } Y] = \Pr_{\mathbf{D}}[(\mathcal{A} \text{ successful in } Y) \wedge \big(|Y_b| \geq (1+\delta)\mu_1 n\big)] + \\
&\quad + \Pr_{\mathbf{D}}[(\mathcal{A} \text{ successful in } Y) \wedge \big(|Y_b| < (1+\delta)\mu_1 n\big)] \leq \\
&\leq \Pr_{\mathbf{D}}[(|Y_b| \geq (1+\delta)\mu_1 n] + \Pr_{\mathbf{D}}[\mathcal{A} \text{ successful in } Y_a \mid |Y_b| < (1+\delta)\mu_1 n] \leq \\
&\leq e^{-\mu_1 n \delta^2/3} + \big(\mu_2 + \epsilon'\big)^{|Y_a|} \leq e^{-\mu_1 n \delta^2/3} + \big(\mu_2 + \epsilon'\big)^{d/2-(1+\delta)\mu_1 n} \leq \\
&\leq 2\min\{\mu_2^{-1}, e\}^{-(d/2-(1+\delta)\mu_1 n)\delta^2/3} + \mathsf{negl}(\lambda).
\end{aligned}
\tag{13}
$$

We conclude that given that events A, F do not occur $\mathcal{A}$ wins the E2E verifiabiilty game $G_{\text{E2E-Ver}}^{\mathcal{A}_1,\mathcal{E},\mathcal{D},d,\theta,\phi}(1^\lambda, n, m, k)$ where the parameters d, θ, ϕ satisfy the constraints in the statement of the theorem only if

$$
\begin{aligned}
&|X \cup Y| = |X \setminus Y| + |Y| \geq d \geq 2(1+\delta)\max\{\kappa_1, \mu_1\}n, \\
&n - (|T_{<}| - |X_{<}| \geq n - |T \setminus X| = |\mathcal{V}_{\mathsf{succ}}| \geq \theta \\
&\qquad \geq n - \big(\frac{1}{(1+\delta)\kappa_2} - 1\big)\big(d/2 - (1+\delta)\kappa_1 n\big), \quad \text{and} \\
&|\mathcal{V}_{\mathsf{comp}}| = \phi \leq (1-\delta)(1-\kappa_3)\big(\frac{1}{(1+\delta)\kappa_2} - 1\big)\big(d/2 - (1+\delta)\kappa_1 n\big).
\end{aligned}
$$

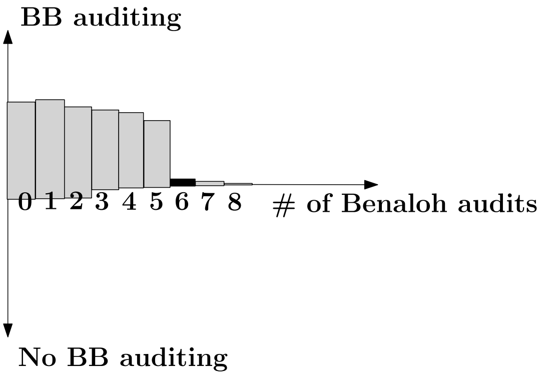


Figure 7: A voter transducer distribution with resistance against VSD and BB attacks ($i^* = 6$, $\mathcal{J} = \{0,1,2,3,4,5\}$).

By Eq. (10),(11),(12),(13), the probabiity that $\mathcal{A}$ wins under the abovve constraints is no more than

$$\begin{aligned}
&\Pr_{\mathbf{D}}[G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1] = \Pr_{\mathbf{D}}\big[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\wedge A\big]+\\
&\quad+\Pr_{\mathbf{D}}\big[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\wedge(\neg A)\big]\leq\\
&\leq(\mu_1+\mu_2-\mu_1\mu_2)^\theta+\mathsf{negl}(\lambda)+\Pr_{\mathbf{D}}[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\wedge\neg F\mid\neg A]\leq\\
&\leq(\mu_1+\mu_2-\mu_1\mu_2)^\theta+\mathsf{negl}(\lambda)+\\
&\quad+\Pr_{\mathbf{D}}\big[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\wedge(|X\setminus Y|\geq d/2\mid(\neg F)\wedge(\neg A)\big]+\\
&\quad+\Pr_{\mathbf{D}}\big[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\wedge(|X\setminus Y|<d/2\mid(\neg F)\wedge(\neg A)\big]\leq\\
&\leq(\mu_1+\mu_2-\mu_1\mu_2)^\theta+\mathsf{negl}(\lambda)+\\
&\quad+\Pr_{\mathbf{D}}\big[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\mid(|X\setminus Y|\geq d/2)\wedge(\neg F)\wedge(\neg A)\big]+\\
&\quad+\Pr_{\mathbf{D}}\big[\big(G^{\mathcal{A},\mathcal{E},d,\theta,\phi}_{\mathrm{E2E-Ver}}(1^\lambda,n,m,k)=1\big)\mid(|Y|\geq d/2)\wedge(\neg F)\wedge(\neg A)\big]\leq\\
&\leq(\mu_1+\mu_2-\mu_1\mu_2)^\theta+\mathsf{negl}(\lambda)+\\
&\quad+3e^{-(1-\kappa_3)\min\left\{\kappa_1 n,\min\{\frac{1}{(1+\delta)\kappa_2}-1,\kappa_2\}\left(d/2-(1+\delta)\kappa_1 n\right)\right\}\delta^2/3}+\\
&\quad+\big(2\min\{\mu_2^{-1},e\}^{-(d/2-(1+\delta)\mu_1 n)\delta^2/3}+\mathsf{negl}(\lambda)\big)\leq\\
&\leq 5\min\{\mu_2^{-1},e\}^{-(1-\kappa_3)\min\{\kappa_1 n,\gamma_2\gamma_1\}\frac{\delta^2}{3}}+(\gamma_3)^\theta+\mathsf{negl}(\lambda),
\end{aligned}$$

where $\gamma_1 = d/2-(1+\delta)\max\{\kappa_1,\mu_1\}n$, $\gamma_2=\min\{\frac{1}{(1+\delta)\kappa_2}-1,\kappa_2\}$ and
$\gamma_3=\mu_1+\mu_2-\mu_1\mu_2$. ∎

Illustrating Theorem 2. To provide intuition, we illustrate an example of a voter transducer distributions that corresponds to conditions (i) and (ii) of Theorem 2 in Figure 7.

4.4 On the tightness of the conditions of Theorems 1 and 2

The conditions stated in Theorems 1 and 2 determine two classes of voter transducer distributions that correspond to vulnerable and insusceptible settings, respectively. We observe that weakening the condition (i) of Theorem 1 (resp. (i) of Theorem 2) cannot imply vulnerability (resp. security). Namely, in condition (i) of Theorem 1, if one of (1.a),(1.b) or (1.c) does not hold, then the adversary cannot be certain that it will achieve a sufficiently large deviation from VSD attacks without increasing rapidly the number of complaints. On the other hand, if condition (i.a) of Theorem 2 does not hold, then E2E verifiability cannot be preserved when (1.b) becomes a disjunction, since a high complaint rate alone is meaningless if the adversary has high success rate of VSD attacks.

Consequently, it is not possible to achieve logical (i.e. probability thresholds are considered either sufficiently ***high*** or sufficiently ***low***) tightness for interesting sets of parameters d, θ, ϕ only by negating the conditions of each of the two theorems. However, this is possible if we assume that the voter's complaining behavior is balanced by flipping coins in order to decide whether they will complain in case of unsuccessful termination, i.e. if we set $\kappa_3 = 1 - \kappa_3 = 1/2$. Specifically, given that $\kappa_3 = 1/2$ is a "neutral" value, we can restate the conditions of Theorems 1 and 2 in their logical form as follows:

Theorem 1: *A voter transducer distribution is* susceptible to VSD or/and BB attacks *if at least one of the following two conditions holds:*

*(i). There is an $i^* \in \{0, \ldots, q\}$ such that (i.a) the probability that a voter executes at least i^* Benaloh audits is **high** AND (i.b) the probability that a voter, given that she has executed at least i^* Benaloh audits, will cast her vote after exactly i^* Benaloh audits is **high**.*

OR

*(ii). There is a subset $\mathcal{J} \subseteq \{0, \ldots, q\}$ such that (ii.a) the probability that a voter executes j Benaloh audits for some $j \in \mathcal{J}$ is **high** AND (ii.b) for every $j \in \mathcal{J}$, the probability that a voter, given she has executed j Benaloh audits, will not audit the BB **high**.*

Theorem 2: *A voter transducer distribution achieves* resistance against VSD and BB attacks *if the following two conditions hold:*

*(i) There is an $i^* \in \{0, \ldots, q+1\}$ such that (i.a) the probability that a voter executes at least i^* Benaloh audits is **low** and (i.b) for every $i \in \{0, \ldots, q\}$, if $i < i^*$, then the probability that a voter, given that she will execute at least i Benaloh audits, will cast her vote after exactly i Benaloh audits is **low**.*

AND

*(ii) There is a subset $\mathcal{J} \subseteq \{0, \ldots, q\}$ such that (ii.a) the probability that a voter executes j Benaloh audits for some $j \in \mathcal{J}$ is **high** AND (ii.b) for every $j \in \mathcal{J}$, the probability that a voter, given she has executed j Benaloh audits, will audit the BB is **high**.*

Based on the above statements, we show that the following hold:

1. *If condition (i) of Theorem 1 does not hold, then condition (i) of Theorem 2 holds*: let $\mathcal{I}_1$ be the set of $i \in \{0, \ldots, q\}$ s.t. the probability that a voter executes at least i Benaloh audits is ***high***. By the negation of condition (i) of Theorem 1, for every $i \in \mathcal{I}_1$, the probability that a voter, given that she will execute at least i Benaloh audits, will cast her vote after exactly i Benaloh audits is ***low***. Observe that $\mathcal{I}_1$ is not empty, as $0 \in \mathcal{I}_1$. Therefore, if we set $i^* = \max\{i \mid i \in \mathcal{I}_1\} + 1$, then, by definition, i^* statisfies the conditions (i.a) and (i.b) of Theorem 2.

2. *If condition (i) of Theorem 2 does not hold, then condition (i) of Theorem 1 holds*: let $\mathcal{I}_2$ be the set of $i \in \{0, \ldots, q+1\}$ s.t. the probability that a voter executes at least i Benaloh audits is ***low***. Clearly, $\mathcal{I}_2$ is non-empty, since $q + 1 \in \mathcal{I}_2$). By the negation of condition (i) of Theorem 2, for every $i \in \mathcal{I}_2$ there

is an $i' < i$ s.t. the probability that a voter, given that she will execute at least i Benaloh audits, will cast her vote after exactly i' Benaloh audits is ***high***. In this case, we set i^* to be this i' that corresponds to the minimum i in $\mathcal{I}_2$ (note that $i^* \geq 0$, since $0 \notin \mathcal{I}_2$). In both cases, i^* satisfies the conditions (i.a) and (i.b) of Theorem 1.

3. *If condition (ii) of Theorem 1 does not hold, then condition (ii) of Theorem 2 holds*: by an averaging argument, there is a $j \in \{0, \dots, q\}$ s.t. the probability that a voter executes j Benaloh audits is at least $1/(q+1)$. Assuming that the maximum number of Benaloh audits q is small (which is meaningful for most interesting cases in practice), we can consider $1/(q+1)$ to be a sufficiently ***high*** probability. By the negation of condition (ii) of Theorem 1, for singleton $\{j\}$, the probability that a voter that executes j Benaloh audits wil audit the BB is ***high***. Thus, the set $\mathcal{J}$ that contains all j for which the voter executes j Benaloh audits with probability at least $1/(q+1)$ satisfies the conditions (ii.a) and (ii.b) of Theorem 2.

4. *The negation of condition (ii) of Theorem 2 implies the condition (ii) of Theorem 1*: by the negation of condition (ii) of Theorem 2, every j for which the voter executes j Benaloh audits with probability at least $1/(q+1)$ (***high***) determines a subset (singleton $\{j\}$) of ***low*** BB auditng probability. Thus, the set $\mathcal{J}$ that contains all j for which the voter executes j Benaloh audits with probability at least $1/(q+1)$ satisfies the conditions (ii.a) and (ii.b) of Theorem 1.

5 Voter Privacy/Receipt-Freeness of an e-voting ceremony

In this Section, we prove the voter privacy/receipt-freeness of the Helios e-voting ceremony. The proof consists of (i) the construction of simulator $\mathcal{S}$ for the voter privacy/receipt-freeness game, and (ii) the reduction showing that any adversary who has non-negligible advantage in the voter privacy/receipt-freeness game can be used to break the IND-CPA security of the underlying ElGamal encryption scheme.

Theorem 3. *Assume an election run of Helios with n voters, m candidates and k trustees. Assume that the hash function $H(\cdot)$ considered in Section 3 is a random oracle. Let $m, n, k, t \in \mathbb{N}$ be polynomial in λ. If the underlying ElGamal encryption scheme is IND-CPA secure, then there exists a* $\mathbf{Cast}$ *ceremony simulator $\mathcal{S}$ s.t. for all distribution collections $\mathcal{D}$ and for all PPT adversary $\mathcal{A}$, the distinguishing advantage of the voter privacy/receipt-freeness game for Helios is*

$$\left|\Pr[G^{\mathcal{A},\mathcal{S},\mathcal{D}}_{t\text{-priv}}(1^\lambda, n, m, k) = 1] - 1/2\right| = \mathsf{negl}(\lambda).$$

Proof. The proof is carried out via a reduction. Namely, we show that if there exists a PPT adversary $\mathcal{A}$ that wins the voter privacy/receipt-freeness game for Helios with non-negligible distinguishing advantage, then there exists a PPT adversary $\mathcal{B}$ that breaks the IND-CPA security of the ElGamal encryption scheme with blackbox access to $\mathcal{A}$. Through the proof, we view $H(\cdot)$ as a random oracle (RO).

The construction of simulator $\mathcal{S}$. Recall that in the execution of the $\mathbf{Cast}$ ceremony, V_ℓ and VSD are controlled by the challenger. V_ℓ behaves according to the sampled transducer $M_{i_\ell, c_\ell, a_\ell} \leftarrow \mathbf{D}_\ell$, which audits the ciphertexts produced by the VSD i_ℓ times before encrypting its real candidate selection. Note that value of c_ℓ, a_ℓ is irrelevant for privacy, as the EA is honest and checks the validity of all the submitted ballots as well as the associated NIZK proofs in the privacy game. For the j-th ciphertext auditing, it sends the VSD the candidate selection $\mathcal{U}^b_\ell$ and obtains the created ballot $\psi_{\ell,j}$ and the corresponding randomness $r_{\ell,j}$ from the VSD. After the j-th auditing, it sends the candidate selection $\mathcal{U}^b_\ell$ to the VSD and casts the created ballot ψ_ℓ together with its identity ID_ℓ. The view of V_ℓ is defined as $\mathcal{I}_\ell = \langle(\mathrm{Pub}, s_\ell, \mathcal{U}^b_\ell), (\psi_{\ell,j}, r_{\ell,j})_{j\in[i_\ell]}, \alpha_\ell\rangle$, where $\alpha_\ell = (\psi^1_\ell, \mathrm{ID}_\ell)$ is the receipt.

The simulator $\mathcal{S}$ randomly picks a coin $b' \leftarrow \{0,1\}$ on its first execution and maintains the coin b' throughout the privacy game. On input $(\mathcal{I}_\ell, \mathcal{U}_\ell^0, \mathcal{U}_\ell^1)$, $\mathcal{S}$ for $j \in \{1, \ldots, i_\ell\}$ creates ballot $\psi'_{\ell,j}$ using a fresh randomness $r'_{\ell,j}$ for the candidate selection $\mathcal{U}_\ell^{b'}$, as VSD would. It then outputs the simulated view $\mathcal{I}'_\ell = \langle(\mathrm{Pub}, s_\ell, \mathcal{U}_\ell^{b'}), (\psi_{\ell,j}, r_{\ell,j})_{j\in[i_\ell]}, \alpha_\ell\rangle$, where $\alpha_\ell = (\psi_\ell^1, \mathrm{ID}_\ell)$ remains the same.

The reduction. Assume that $\mathcal{A}$ is a PPT adversary that wins the voter privacy/receipt-freeness game $G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}(1^\lambda, m, n, k)$, for some $m, t, n, k \in \mathbb{N}$ polynomial in λ, We construct an adversary $\mathcal{B}$ that tries to use $\mathcal{A}$ in a blackbox manner to attack the IND-CPA security of the ElGamal encryption. As shown in [BPW12], strong Fiat-Shamir transformations of Σ protocols are simulation sound extractable. More specifically, for any prover $\mathcal{A}$ who outputs polynomially many statement/proof pairs $(\vec{Y}, \vec{\Pi})$, there exists an efficient knowledge extractor $\mathcal{K}$, given black-box access to $\mathcal{A}$ and may invoke further copies of $\mathcal{A}$ using the same randomness as was used in the main run, can extract a vector of witnesses $\vec{w}$ corresponding to the statements $\vec{Y}$. Consider the following sequence of games from G_0 to G_3.

Game G_0: The actual game $G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}(1^\lambda, n, m, k)$, where the challenger uses $\mathcal{U}_\ell^b$ in the **Cast** ceremony and the above simulator $\mathcal{S}$ is invoked when $b = 1$.

Game G_1: Game G_1 is the same as Game G_0 except the following. The challenger $\mathcal{C}$ controls the RO $H(\cdot)$. After the **Cast** phase, $\mathcal{C}$ invokes the knowledge extractor $\mathcal{K}$ to extract the partial secret keys $\{\mathsf{sk}_i\}_{i\neq w}$ of all the other trustees that $\mathcal{A}$ controls and the candidate selections of all the casted ballots submitted by the corrupted voters. The challenger $\mathcal{C}$ aborts if the extraction fails; otherwise, $\mathcal{C}$ completes the experiment.

Game G_2: Game G_2 is the same as Game G_1 except the following. The challenger $\mathcal{C}$ computes the election result $\langle x_1, \ldots, x_m\rangle$ that corresponds to the ballots that $\mathcal{A}$ posted on the BB according to the candidate selections of the corrupted voters extracted in Game G_1. Denote the final tally ElGamal ciphertext vector as $\langle C_1, \ldots, C_m\rangle$, where $C_j := (C_j^{(0)}, C_j^{(1)}) = (g^{r_j}, g^{x_j} \cdot h^{r_j})$ for some r_j. For $j \in \{1, \ldots, m\}$, the trustee T_w produces its partial decryption of C_j as $D_{w,j} = C_j^{(1)}/(g^{x_j} \cdot (C_j^{(0)})^{\sum_{i\neq w} \mathsf{sk}_i})$ together with simulated NIZK proofs without using its partial secret key.

Game G_3: Game G_3 is the same as Game G_2 except the following. For all the voters $V_\ell \in \tilde{\mathcal{V}}$, the challenger $\mathcal{C}$ submits a vector of encryptions of 0 together with the simulated NIZK proof instead of the real ciphertexts of the candidate selections. Besides, the challenger $\mathcal{C}$ always give the adversary $\mathcal{A}$ the simulated **Cast** views, ignoring the bit b.

Define $\mathsf{Adv}_{G_i,G_j}(\mathcal{A}) := \frac{1}{2}\big|\Pr[\mathcal{A} = 1 \mid G_i] - \Pr[\mathcal{A} = 1 \mid G_j]\big|$. We complete the proof by showing a sequence of indistinguishability claims for the games G_0, G_1, G_2, G_3.

– *G_0 is indistinguishable from $G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}(1^\lambda, n, m, k)$:* by definition of the the voter privacy/receipt-freeness game, $\big|\Pr[G_{t\text{-priv}}^{\mathcal{A},\mathcal{S},\mathcal{D}}(1^\lambda, n, m, k) = 1] - \Pr[\mathcal{A} = 1 \mid G_0]\big| = 0$.

– *G_1 is indistinguishable from G_0:* the probability that the knowledge extractor fails to extract the witnesses is negligible . Upon successful extraction, the view of $\mathcal{A}$ is identical to G_0. Hence, we have $\mathsf{Adv}_{G_0,G_1}(\mathcal{A}) = \mathsf{negl}(\lambda)$.

– *G_2 is indistinguishable from G_1:* since the simulated NIZK proofs are identical to the real ones, the view of $\mathcal{A}$ is identical to G_1. Hence, we have $\mathsf{Adv}_{G_1,G_2}(\mathcal{A}) = 0$.

– G_3 *is indistinguishable from* G_2*:* it is easy to see that the tally ciphertexts will still be decrypted to the correct election result $\langle x_1, \ldots, x_m \rangle$ due to the fake partial decryptions $D_{w,j}$. The simulated NIZK proofs are indistinguishable from the real ones.

We now show that if the adversary $\mathcal{A}$ can distinguish Game G_3 from G_2 then there exists an adversary $\mathcal{B}$ who can win the IND-CPA game of the ElGamal encryption with the same probability.

In the IND-CPA game, $\mathcal{B}$ first receives a public key denoted as (g, h_w) from the IND-CPA challenger, and $\mathcal{B}$ forwards (g, h_w) together with the simulated NIZK to the EA as the partial public key of the trustee T_w in the **Setup** phase. Then $\mathcal{B}$ submits $m_0 = 0, m_1 = 1$ to the IND-CPA challenger, and $\mathcal{B}$ receives $C := (C^{(0)}, C^{(1)})$ that encrypts m_{b^*}, where $b^* \in \{0,1\}$ is the IND-CPA challenger bit for $\mathcal{B}$ to guess. $\mathcal{B}$ computes $\hat{C} := (\hat{C}^{(0)}, \hat{C}^{(1)}) = (C^{(0)}, C^{(1)} \cdot (C^{(0)})^{\sum_{i \neq w} \mathsf{sk}_i})$, which is encryption of m_{b^*} under the election public key (g, h). During the **Cast** ceremony, for each uncorrupted voter V_ℓ, $\mathcal{B}$ sets j^*_ℓ to be the index s.t. $\{P_{j^*_\ell}\} = \mathcal{U}^b_\ell$. Then, it generates $m-1$ encryptions of 0, $\{C_{\ell,i}\}_{i \neq j^*_\ell}$ under the election public key (g, h) together with their NIZK. For j^*_ℓ, $\mathcal{B}$ sets C_{ℓ,i^*_ℓ} to be re-encryption of $\hat{C}$, i.e. $C_{\ell,i^*_\ell} = (\hat{C}^{(0)} \cdot g^{r_j}, \hat{C}^{(1)} \cdot h^{r_j})$ for fresh randomness r_j. $\mathcal{B}$ appends necessary simulated NIZK and submits $\{C_{\ell,i}\}_{i \in [m]}$ as the ballot for V_ℓ. Clearly, if C encrypts 0 then the adversary $\mathcal{A}$'s view is the same as Game G_3; otherwise, if C encrypts 1 then the adversary $\mathcal{A}$'s view is the same as Game G_2. Hence, assume $\mathcal{A}$ outputs 1 if she thinks she is in Game G_2 and outputs 0 if she thinks she is in Game G_3. $\mathcal{B}$ forwards $\mathcal{A}$'s outputs, and $\mathcal{B}$ win the IND-CPA game whenever $\mathcal{A}$ guesses correctly. Thus, we have $\mathsf{Adv}_{G_2,G_3}(\mathcal{A}) = \mathsf{Adv}^{\text{IND-CPA}}_{\text{ElGamal}}(\mathcal{A}) = \mathsf{negl}(\lambda)$.

– $\Pr[\mathcal{A} = 1 \mid G_3] = 1/2$*:* since the view of Game G_3 does not depend on the bit b, the adversary's probability of guessing b correctly in G_3 is exactly $1/2$.

By the above claims, the overall advantage of $\mathcal{A}$ is

$$\begin{aligned}
\left|\Pr[G^{\mathcal{A},\mathcal{S},\mathcal{D}}_{t\text{-priv}}(1^\lambda, n, m, k) = 1] - \frac{1}{2}\right| &= \left|\Pr[\Pr[\mathcal{A} = 1 \mid G_0] - \Pr[\mathcal{A} = 1 \mid G_3]\right| \leq \\
&\leq \sum_{i=1}^{3} \mathsf{Adv}_{G_{i-1},G_i}(\mathcal{A}) = \\
&= \mathsf{negl}(\lambda) + 0 + \mathsf{Adv}^{\text{IND-CPA}}_{\text{ElGamal}}(\mathcal{A}) = \\
&= \mathsf{negl}(\lambda),
\end{aligned}$$

which completes the proof. ■

6 Evaluating the E2E verifiability of an e-voting ceremony

In this section, we evaluate our results for the E2E verifiability of Helios, by instantiating the bounds in Theorems 1 and 2 for various voter transducer distributions. Our evaluations are separated into two categories: (i) evaluations that are based on actual human data that derive from elections using Helios and (ii) evaluations that are based on simulated data for various sets of parameters.

6.1 Evaluations based on human data.

Our human data are sampled from two independent surveys: the first sample is from the member elections of the Board of Directors of the International Association for Cryptographic Research (IACR);

	Benaloh audits							
	0		1		2		3	
BB audit	Yes	No	Yes	No	Yes	No	Yes	No
	2	22	4	5	1	0	1	0

Table 1: Distribution of the voters' VSD and BB auditing behavior in the IACR sample consisting of 35 responders.

the second is a non-binding poll among the students of the Department of Informatics and Telecommunications (DI&T) of the University of Athens. In the following section, we present at length our methodology for the two surveys.

6.1.1 Methodology of our surveys with human subjects

The methodology for IACR elections. We conducted our survey using the SurveyMonkey tool. Specifically, we formed a questionnaire that consisted of three questions, as shown in Figure 8.

QUESTIONNAIRE

Q1. In the last IACR election you participated, did you use the *"audit" your ballot* functionality (where you get to see the opening of the ciphertext containing your vote)?

Yes: ☐ **No:** ☐

Q2. If you answered "**Yes**" in the above question, how many times did you audit?

Enter a positive integer: ☐

Q3. Did you verify that the smart ballot tracker (the hash of your submitted ciphertext) was actually posted on the ballot tracking center (the public web-site that lists all encrypted ballots)?

Yes: ☐ **No:** ☐

Figure 8: The questionnaire used in the survey on the voter's behavior at the IACR elections.

The questionnaire was delivered to the IACR board. In turn, the board sent an open call to the IACR members for volunteering to participate in our survey. By the end of the survey, we collected 35 responses, from which we extracted the data presented in Table 1.

The methodology for DI&T poll. We conducted a non-binding poll among the students of the DI&T Department of the University of Athens. During a lecture of the Computer Security course, we gave a presentation of Helios, focusing on the importance of auditing their ballots. Then, we asked the students to participate in an election run using Helios which concept concerned the improvement of their daily student life. Specifically, the survey consisted of two stages; in the first stage, the students had a period of one week prior to the election to form a proposal that would reply to the following question:

Given a €10,000 budget, which department facility
would you suggest that should be updated or developed?

Benaloh audits		
0	1	2
20	27	2

Table 2: Distribution of the voters' VSD auditing behavior at the DI&T poll. The sample consists of 49 participants.

In the second stage, at the voting phase, all the submitted proposals where considered as options for the above question. In detail, the question as shown in the Helios booth template is depicted in Figure 9.

QUESTION

Given a € 10,000 budget, which department facility would you suggest that should be updated or developed?

Select up to 2 options:

1. Improving WiFi coverage in all areas of the department building complex. ☐

2. Extension of night lighting in all external areas of the building complex. ☐

3. Printer room with off-hours student access. ☐

4. Extended access to student reading room via card based gate access control. ☐

Figure 9: The question template at the DI&T poll.

A total of 49 students participated in our survey. We modified the Helios codebase so that our server could track the auditing behavior of the participants. The data extracted from the voting process are presented in Table 2.

Parameter computation. The parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2$ used in Theorem 1 express the vulnerability of Helios voting ceremony against verifiability attacks w.r.t. a specific voter transducer distribution. It is easy to see that every $i^* \in \{0, \ldots, q\}$ and $\mathcal{J} \subseteq \{0, \ldots, q\}$ (where q is the maximum number of Benaloh audits) imply a set of parameters $(\kappa_1, \kappa_2, \kappa_3)$ and (μ_1, μ_2) that determine the success probability of an attacker against the VSD vulnerability and the BB vulnerability when the voter executes i^* and $j \in \mathcal{J}$ Benaloh audits respectively. The formulas and the security significance of parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2$ is explained in Table 3.

By Table 3, we can deduce that parameters $\kappa_1, \kappa_3, \mu_1$ determine the *size* of the subsets of vulnerable voters, while κ_2, μ_2 can be seen as measures of the *quality* of the VSD and BB attacks.

In order to evaluate the vulnerability of the voter behavior in each survey we performed the following procedure:

- We focused on maximizing the success probability that each type of attack may be mounted leaving the parameters d, θ, ϕ as free variables[5].

[5]Following a different approach, one could also consider optimizing all parameters simultaneously including d, θ, ϕ. Performing such analysis could be interesting future work; nevertheless, our analysis already reveals significant security deficiencies in our experiments.

Parameter	Formula for the parameter	Security Significance
κ_1	$\Pr\left[\bigvee_{\substack{0\le i<i^* \\ c,a\in\{0,1\}}} E_{i,c,a}\right]$	As κ_1 decreases, the guarantee that the voter will execute at least i^* audits increases.
κ_2	$\Pr\left[\bigvee_{c,a\in\{0,1\}} E_{i^*,c,a} \mid \bigvee_{\substack{0\le i<i^* \\ c,a\in\{0,1\}}} E_{i,c,a}\right]$	As κ_2 decreases, the success rate of a VSD attack after the i^*-Benaloh audit increases.
κ_3	$\Pr\left[E_{i^*,0,0} \vee E_{i^*,0,1}\right]$	As κ_3 decreases, the complaint rate due to failed VSD attacks after the i^*-Benaloh audit increases.
μ_1	$\Pr\left[\bigvee_{\substack{j\notin\mathcal{J} \\ c,a\in\{0,1\}}} E_{j,c,a}\right]$	As μ_1 decreases, the rate of voters that "fall" into the target subset $\mathcal{J}$ increases.
μ_2	$\max_{j\in\mathcal{J}}\left\{\Pr\left[E_{j,0,1} \vee E_{j,1,1}\right]\right\}$	As μ_2 decreases, the success rate of a BB attack against a voter that "falls" into the target subset $\mathcal{J}$ increases.

Table 3: The formula and the security significance of parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2$ used in Theorem 1 for given $i^* \in \{0, \ldots, q\}$ and $\mathcal{J} \subseteq \{0, \ldots, q\}$, where q is the maximum number of Benaloh audits. $E_{i,c,a}$ is the event that voter's behavior follows the transducer $M_{i,c,a}$.

- For both surveys, no complaints or audit failures were reported. Hence, due to lack of data, we choose a "neutral" value for κ_3 equal to 0.5 (see also Section 4.4). Note that our analysis will hold for any other *not close to* 0 value of κ_3. The case of $\kappa_3 = 0$, i.e., when the voter always complains to the authority when a Benaloh audit goes wrong, would make VSD attacks unattractive in the case that ϕ is small and would suggest that the attacker will opt for BB attacks (if such attacks are feasible which depends on μ_1, μ_2).

- For both surveys, we ran an exhaustive search in all possible numbers of Benaloh audits to locate the index i^* s.t. the parameters κ_1, κ_2 that maximize the probability of success stated in Theorem 1:*condition (i)*. Equivalently, we searched for the values κ_1, κ_2 that maximize the function

  $$F_\delta(\kappa_1, \kappa_2) = (1 - \kappa_1)(\kappa_2 - \delta + \delta\kappa_2)(1 - \kappa_2)$$

  for a suitably small value of $\delta \in [0, 1)$.

- For the IACR survey, we ran exhaustive search in all subsets of $\{0, 1, 2\}$ to locate the subset $\mathcal{J}$ s.t. the parameters μ_1, μ_2 that maximize the probability of success stated in Theorem 1:*condition (ii)*, lower bounded by the equation

  $$(1 - e^{-(1-\mu_1)n\frac{\delta^2}{2}})(1 - \mu_2)^d, \quad \text{where } \delta \in [0, 1).$$

  Since the probability bound drops exponentially as the tally deviation d increases, the term $(1 - e^{-(1-\mu_1)n\frac{\delta^2}{2}})$ quickly becomes insignificant as compared with the term $(1 - \mu_2)^d$. Consequently, we concentrated on the asymptotic behavior of the equation by searching for the minimum μ_2 that leads to a slower decreasing rate.

Following the above procedure, we computed the optimal (from an adversarial point of view) sets of parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2$ as shown in Table 4.

Survey	i^*	$\mathcal{J}$	Parameters				
			κ_1	κ_2	κ_3	μ_1	μ_2
IACR elections	0	$\{0\}$	0	0.315	0.5	0.315	0.084
DI&T poll	1	–	0.408	0.069	0.5	–	–

Table 4: Instantiated parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2$ of Theorem 1 for both surveys.

6.2 Analysis of the experiments

Analysis of the IACR survey. From the first row of Table 4, we read that $\mu_2 = 0.084$ which is a very small value as opposed to $\kappa_2 = 0.315$. Thus, we expect that elections where the electorate follows the voter transducer distribution of IACR elections are much more vulnerable to BB attacks rather than VSD attacks. Indeed, this is consistent with the analysis that we describe below.

We computed the percentage of *tally deviation/No. of voters* that the adversary can achieve when the success probability is lower bounded by 25%, 10%, 5% and 1% for various electorate scales. Specifically, we observed that the success probability bounds stated in Theorem 1 express more accurately the effectiveness of the adversarial strategy for (i) medium to large scale elections when the adversary attacks via the VSD and (ii) for small to medium scale elections when the adversary attacks via the BB. As a consequence, we present our analysis for $n = 100, 500, 1000, 2500$ and 5000 voters w.r.t. BB attack effectiveness and for $n = 5000, 10000$ and 50000 voters w.r.t. VSD attack effectiveness. Our findings are shown in the tables in Tables 5 and 7.

Voters	Success probability %			
	≥ 25	≥ 10	≥ 5	≥ 1
100	15.92	26.4	34.42	51.42
500	3.18	5.28	6.87	10.56
1000	1.59	2.64	3.42	5.28
2500	0.636	1.05	1.37	2.11
5000	0.31	0.52	0.68	1.05

Table 5: Percentage of *tally deviation/No. of voters* achieved in elections under BB attack strategies against electorates following the voter transducer distribution of IACR elections. The attack succeeds even when $\theta = n$ and $\phi = 0$.

The data in Table 5 illustrate the power of BB attacks against compact bodies of voters (e.g. organizations, unions, board elections, etc.) where BB auditing is rare. We can see that in the order of hundreds more than 5% of the votes could be swapped with significant probability of no detection. This power deteriorates rapidly as we enter the order of thousands, however, the election result could still be undermined, as deviation between 1%-2%, is possible, without the risk of *any* complaint due to unsuccesful engagement in the Cast ceremony (i.e. $\theta = n$ and $\phi = 0$). Therefore, even in a setting of high complaint rate (κ_3 is close to 0), the adversary may turn into a BB attack strategy and still be able to alter radically the election result, as marginal differences are common in all types of elections. We stress that from published data we are aware of[6], there have been elections for the IACR board where the votes for winning candidates were closer than 3% to the votes of candidates that lost in the election. Therefore, if the voter distribution had been as the one derived by Table 4, and 500 members had voted, the result could have been overturned with success probability 25% even if a single complaint was considered to

[6]For instance, in the IACR elections of 2013, (cf. `https://www.iacr.org/elections/2013/`) the cutoff between the last elected director and the first candidate that was not elected was less than 3%.

Year	Participants	Cutoff %	Success probability %
2015	437	6.87	7.35
2014	575	5.57	6.17
2013	637	2.99	19.14
2012	518	11.59	0.5
2011	621	4.03	11.35
2010	475	8.64	2.82
2009	325	4.93	24.8
2008	312	0.33	91.66
2007	–	–	–
2006	324	4.33	29.57

Table 6: Success probability of a hypothetical BB attack strategy against the IACR elections for the Board of Directors per election year. The success probability is computed given the number of participants and the cutoff between the last elected director and the first candidate that was not elected. The dashed line denotes the actual start of Helios use for IACR elections. Regarding the year 2007, no data were recorded in `https://www.iacr.org/elections/`.

Success probability %	n=5000			n=10000			n=50000		
	d/n	θ/n	ϕ/n	d/n	θ/n	ϕ/n	d/n	θ/n	ϕ/n
≥ 25	51.8	54.3	25.7	57.5	59.6	21.9	63.9	65.0	18.1
≥ 10	52.8	55.3	25.1	58.1	60.2	21.5	64.1	65.2	18.0
≥ 5	53.2	55.6	24.8	58.3	60.3	21.4	64.2	65.2	17.9
≥ 1	53.4	55.9	24.7	58.4	60.5	21.3	64.3	65.3	17.8

Table 7: Effectiveness of VSD attack strategies against electorates with $n = 5000, 10000$ and 50000 voters following the voter transducer distribution in IACR elections. In the tables, d/n is the percentage of *tally deviation/No. of voters*, θ/n is the ratio of honest successful voters in % and ϕ/n is the ratio of honest complaining voters in %.

be a "stop election event" (since $\phi = 0$).

To provide more context, in Table 6, we provide the cutoff between elected and non-elected candidates for the last 10 years of IACR elections for the Board of Directors, followed by the exact success probability of a hypothetical BB attack strategy to overturn the election result given the actual number of cast ballots per year. We observe that the attacker success probability for many of the elections is considerable.

On the other hand, as already mentioned, the effectiveness of VSD attacks is clear if we scale the electorate in the order of thousands and above. As we see from the results in Table 7, a VSD attack strategy against an election that follows the voter distribution in IACR elections would not have a great impact unless an unnatural number of complaints could be tolerated. Indeed, even for the scale of 50000 voters, the rate of complaints that is ignored must be close to 17% which is rather unacceptable in a real world setting (such number of complaints would most definitely lead to a stop election event).

We conclude that the IACR voter behavior is susceptible to BB attacks with significant probability of success but not VSD attacks unless there is high tolerance in voter complaints.

Analysis of the DI&T poll. From the second row of Table 4, we read that $\kappa_2 = 0.069$ which is a very small value. Therefore, we expect that voters' behavior in DI&T poll will be vulnerable to VSD attacks.

Success probability %	d/n	θ/n	ϕ/n
≥ 25	52.87	94.67	27.28
≥ 10	53.00	94.75	26.76
≥ 5	53.04	94.77	26.63
≥ 1	53.07	94.79	26.53

Table 8: Effectiveness of VSD attack strategies against electorates with $n = 100000$ voters following the voter transducer distribution of elections DI&T poll. The table notation $d/n, \theta/n, \phi/n$ is as in Table 7.

Our results are presented in Table 8.

It is easy to see that the data in Table 8 add to the intuition on the power of the VSD attacks. One may observe that a very small value of $\kappa_2 = 0.069$ for election DI&T poll leads to efficient attacks while keeping a very high rate of honest voters ($\approx 95\%$), as compared with the cases for elections IACR elections ($\approx 65\%$) where $\kappa_2 = 0.315$.

In the analysis of Table 8, we scaled to 100000 voters so that the probability bound in Theorem 1 reveals the effectiveness of the VSD attacker. Of course, this does not mean that a medium scale election where the probability of a successful VSD attack is $1 - \kappa_2 = 93.1\%$ is not assailable. For instance, consider an electorate of $n = 500$ voters following the transducer distribution of the DI&T poll and a VSD attacker as the one described in the proof of Theorem 1. It easy to show that the attacker can achieve tally deviation $\beta\%$ without *any* complaint (i.e., $\theta = n$ and $\phi = 0$ as in a BB attack strategy) with probability at least

$$(1 - e^{-(1-\kappa_1)n\frac{\delta^2}{2}})(1 - \kappa_2)^{\beta n} = (1 - e^{-148\delta^2})(0.931)^{500\beta}, \tag{14}$$

for $d \leq (1 - \delta)296$ and any $\delta \in [0, 1)$. In Table 9, we present the ratio of tally deviation achieved by the attacker for various success probabilities, as derived from Eq. (14). Observe that tally deviation 5% may occur with 16.7% probability, which is certainly significant and reveals VSD vulnerability even at medium scale elections.

Success probability %			
≥ 25	≥ 10	≥ 5	≥ 1
0.013	2.8	16.7	69.9

Table 9: Percentage of *tally deviation*/*No. of voters* achieved in elections under VSD attack strategies against electorates of 500 voters following the voter transducer distribution of DI&T poll. The attack succeeds even when $\theta = n$ and $\phi = 0$.

We conclude that the DI&T voter behavior is susceptible to VSD attacks with significant probability. We cannot draw a conclusion for BB attacks since we did not collect auditing data for this case.

6.3 Evaluations based on simulated data.

Our human data analysis is obtained by real bodies of voters that have an imperfect voting behavior. To understand what would be the security level of a Helios e-voting ceremony when executed by an "ideal" (perfectly trained) electorate, we ran simulations of election executions. From a modeling point of view, the voters' behavior tends to become "ideal", when the parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2$ in Theorem 2 become smaller. In our simulations, we estimated the security that Theorem 2 can guarantee, considering various parameter values from 0.25 to 0.03125. In addition, we ran our computations by

d/n	**Detection Probability %**	θ/n	ϕ/n
7.8	57.4	100.0	0.00
9.8	77.1	75.9	17.4
11.7	87.6	51.9	34.9
13.6	93.3	27.9	52.3
15.6	96.4	3.9	69.0

Table 10: Detection probability of *(tally deviation)/(No. of voters)* percentage for elections with $n = 100000$ voters for simulation parameters $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2 = 0.03125$. The detection probability is defined as $(1 - \epsilon) \cdot 100\%$, where ϵ is the error stated in Theorem 2. The table notation d/n, θ/n, ϕ/n is as in Table 7.

fluctuating the number of voters and the success/error probability. We deduce that even when the election experiment is simulated for a voter transducer distribution with a seemingly good set of parameters (e.g. $\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2 = 0.03125$) and for large scale elections (e.g. 100000 voters), the percentage of tally deviation that could be *guaranteed* to be detected with significant detection probability (e.g. 77%) was relatively big (e.g. 9.8%).

Our conclusion is that the voters must behave almost ideally ($\kappa_1, \kappa_2, \kappa_3, \mu_1, \mu_2 \longrightarrow 0$) in order for a high level of security to be achieved. We present our evaluation in the table in Table 10.

7 Conclusions

In this work we initiated the study of e-voting ceremonies as an extension of traditional security modeling and analysis of e-voting systems. Our framework includes the human participants explicitly as nodes of the protocol and treats them as probability distributions over a set of admissible behaviors modeled as transducers. We argue that this captures more effectively the notion of verifiability since the correctness of the tally is *impossible to be verified* without taking into account the behavior of the voters as a whole.

We applied our framework in the analysis of Helios which is currently the most widely used publicly available e-voting system that offers an end-to-end verifiability mechanism. The behavior of a human node when interacting with the Helios system as a voter includes participation in the cast-or-audit phase provided by the voting booth application of the system as well as the auditing (or not) of the "ballot-tracker" string against the published data in the bulletin board. Within our framework, we characterize the class of voter behaviors under which verifiability may collapse as well as the complementary class of behaviors under which verifiability is upheld.

We collected data from human subjects with the purpose of comparing them with the classes of distributions that we have identified and we concluded, in two different experiments, that the observed behaviors were not consistent with high confidence level in the election results. As a matter of fact, in particular instances, election results could have been overturned with probability as high as 25% without being detected.

We hope that our work will motivate further research in the safe deployment of e-voting systems in real world elections and promote more responsible voter behavior. Also, viewing an e-voting system as a ceremony introduces the set of admissible voter behaviors as a parameter of the system, and hence one may seek to optimize the design towards the simplest possible sets of admissible behaviors (or those that are the most favorable in terms of being implemented by actual humans) that are consistent with security.

References

[Adi08] Ben Adida. Helios: Web-based open-audit voting. In *USENIX Security Symposium*, 2008.

[AOZZ15] Joël Alwen, Rafail Ostrovsky, Hong-Sheng Zhou, and Vassilis Zikas. Incoercible multi-party computation and universally composable receipt-free voting. In *CRYPTO*, 2015.

[BCP+11] David Bernhard, Véronique Cortier, Olivier Pereira, Ben Smyth, and Bogdan Warinschi. Adapting helios for provable ballot privacy. In *ESORICS*, 2011.

[Ben87] Josh Benaloh. Verifiable secret-ballot elections. Yale University Ph.D. Thesis YALEU/DCS/TR-561. New Haven, CT, 1987.

[Ben06] Josh Benaloh. Simple verifiable elections. In Dan S. Wallach and Ronald L. Rivest, editors, *EVT*. USENIX Association, 2006.

[BPW12] David Bernhard, Olivier Pereira, and Bogdan Warinschi. How not to prove yourself: Pitfalls of the fiat-shamir heuristic and applications to helios. In *ASIACRYPT*, 2012.

[BT94] Josh Cohen Benaloh and Dwight Tuinstra. Receipt-free secret-ballot elections (extended abstract). In *STOC*, 1994.

[CEC+08] David Chaum, Aleks Essex, Richard Carback, Jeremy Clark, Stefan Popoveniuc, Alan Sherman, and Poorvi Vora. Scantegrity: End-to-end voter-verifiable optical-scan voting. *Security & Privacy, IEEE*, 6(3):40–46, 2008.

[CF85] Josh D. Cohen and Michael J. Fischer. A robust and verifiable cryptographically secure election scheme (extended abstract). In *FOCS*, 1985.

[CFSY96] Ronald Cramer, Matthew K. Franklin, Berry Schoenmakers, and Moti Yung. Multi-autority secret-ballot elections with linear work. In *EUROCRYPT*, 1996.

[CGS97] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. A secure and optimally efficient multi-authority election scheme. In *EUROCRYPT*, 1997.

[Cha81] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–88, 1981.

[Cha01] David Chaum. Surevote: Technical overview. In *Proceedings of the Workshop on Trustworthy Elections*, WOTE, Aug. 2001.

[Cha04] David Chaum. Secret-ballot receipts: True voter-verifiable elections. *IEEE Security & Privacy*, 2(1):38–47, 2004.

[CMFP+10] Benoît Chevallier-Mames, Pierre-Alain Fouque, David Pointcheval, Julien Stern, and Jacques Traoré. On some incompatible properties of voting schemes. In *Towards Trustworthy Elections*, 2010.

[Com05] United States Election Assistance Commission. Voluntary voting systems guidelines, 2005.

[CP12] Marcelo Carlomagno Carlos and Geraint Price. Understanding the weaknesses of human-protocol interaction. In *Financial Cryptography and Data Security*, 2012.

[CS10] Véronique Cortier and Ben Smyth. Attacking and fixing helios: An analysis of ballot secrecy. *ePrint Archive*, 2010:625, 2010.

[DGK+14] Alex Delis, Konstantina Gavatha, Aggelos Kiayias, Charalampos Koutalakis, Elias Nikolakopoulos, Mema Roussopoulou, Georgios Sotirellis, Panos Stathopoulos, Lampros Paschos, Pavlos Vasilopoulos, Thomas Zacharias, and Bingsheng Zhang. Pressing the button for European elections 2014: Public attitudes towards verifiable e-voting in Greece. In *EVOTE*, 2014.

[DGS03] Ivan Damgård, Jens Groth, and Gorm Salomonsen. The theory and implementation of an electronic voting system. In Dimitris Gritzalis, editor, *Secure Electronic Voting*, volume 7 of *Advances in Information Security*, pages 77–98. Springer, 2003.

[DKR09] Stéphanie Delaune, Steve Kremer, and Mark Ryan. Verifying privacy-type properties of electronic voting protocols. *Journal of Computer Security*, 17(4):435–487, 2009.

[Ell07] Carl M. Ellison. Ceremony design and analysis. *IACR Cryptology ePrint Archive*, 2007:399, 2007.

[Gro04] Jens Groth. Evaluating security of voting schemes in the universal composability framework. In Markus Jakobsson, Moti Yung, and Jianying Zhou, editors, *ACNS*, volume 3089 of *Lecture Notes in Computer Science*, pages 46–60. Springer, 2004.

[JCJ02] Ari Juels, Dario Catalano, and Markus Jakobsson. Coercion-resistant electronic elections. *IACR Cryptology ePrint Archive*, 2002:165, 2002.

[JCJ05] Ari Juels, Dario Catalano, and Markus Jakobsson. Coercion-resistant electronic elections. In Vijay Atluri, Sabrina De Capitani di Vimercati, and Roger Dingledine, editors, *WPES*, pages 61–70. ACM, 2005.

[KKW06] Aggelos Kiayias, Michael Korman, and David Walluck. An internet voting system supporting user privacy. In *ACSAC*, pages 165–174. IEEE Computer Society, 2006.

[KRS10] Steve Kremer, Mark Ryan, and Ben Smyth. Election verifiability in electronic voting protocols. In *ESORICS*, pages 389–404, 2010.

[KTV10a] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Accountability: Definition and relationship to verifiability. *IACR Cryptology ePrint Archive*, 2010:236, 2010.

[KTV10b] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. A game-based definition of coercion-resistance and its applications. In *CSF*, pages 122–136. IEEE Computer Society, 2010.

[KTV11] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Verifiability, privacy, and coercion-resistance: New insights from a case study. In *IEEE Symposium on Security and Privacy*, pages 538–553. IEEE Computer Society, 2011.

[KTV12] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Clash attacks on the verifiability of e-voting systems. In *IEEE Symposium on Security and Privacy*, pages 395–409. IEEE Computer Society, 2012.

[KTW09] Chris Karlof, J. Doug Tygar, and David Wagner. Conditioned-safe ceremonies and a user study of an application to web authentication. In *NDSS*, 2009.

[KZZ15] Aggelos Kiayias, Thomas Zacharias, and Bingsheng Zhang. End-to-end verifiable elections in the standard model. In *EUROCRYPT*, 2015.

[MN06] Tal Moran and Moni Naor. Receipt-free universally-verifiable voting with everlasting privacy. In *CRYPTO*, pages 373–392, 2006.

[Nef04] C. Andrew Neff. Practical high certainty intent verification for encrypted votes. Votehere, Inc. whitepaper, 2004.

[OBV13] Maina M. Olembo, Steffen Bartsch, and Melanie Volkamer. Mental models of verifiability in voting. In James Heather, Steve A. Schneider, and Vanessa Teague, editors, *VOTE-ID*, volume 7985 of *Lecture Notes in Computer Science*, pages 142–155. Springer, 2013.

[PKRV10] Stefan Popoveniuc, John Kelsey, Andrew Regenscheid, and Poorvi Voral. Performance requirements for end-to-end verifiable elections. EVT/WOTE, 2010.

[RBNB11] Kenneth Radke, Colin Boyd, Juan Manuel González Nieto, and Margot Brereton. Ceremony analysis: Strengths and weaknesses. In *IFIP*, 2011.

[SFC] Ben Smyth, Steven Frink, and Michael R. Clarkson. Computational election verifiability: Definitions and an analysis of helios and jcj. Technical report.

[SK95] Kazue Sako and Joe Kilian. Receipt-free mix-type voting scheme - a practical solution to the implementation of a voting booth. In *EUROCRYPT*, 1995.

[TPLT13] Georgios Tsoukalas, Kostas Papadimitriou, Panos Louridas, and Panayiotis Tsanakas. From helios to zeus. In *EVT/WOTE*. USENIX Association, 2013.

[UMQ09] Dominique Unruh and Jörn Müller-Quade. Universally composable incoercibility. *IACR Cryptology ePrint Archive*, 2009:520, 2009.

[ZCC$^+$13] Filip Zagórski, Richard Carback, David Chaum, Jeremy Clark, Aleksander Essex, and Poorvi L. Vora. Remotegrity: Design and use of an end-to-end verifiable remote voting system. In *ACNS*, 2013.