Security models for everlasting privacy ⋆

Panagiotis Grontas, Aris Pagourtzis, and Alexandros Zacharakis

School of Electrical and Computer Engineering
National Technical University of Athens
pgrontas@corelab.ntua.gr, pagour@cs.ntua.gr, azach@corelab.ntua.gr

Abstract. We propose security models for everlasting privacy, a property that protects the content of the votes cast in electronic elections against future and powerful adversaries. Initially everlasting privacy was treated synonymously with information theoretic privacy and did not take advantage of the information available to the adversary and his behavior during or after the election. More recent works provided variations of the concept, limiting the view of the future adversary to publicly available data. We consider an adversary that potentially has insider access to private election data as well. We formally express our adversarial model in game based definitions build on top of a generic voting scheme. This allows us to define a stronger version of everlasting privacy and contrast the two main proposals to achieve it, namely perfectly hiding commitment schemes and anonymous channels.

Keywords: Electronic voting, everlasting privacy, perfectly hiding commitment schemes, blind signatures, anonymous channels

1 Introduction

Electronic voting is not simply a digital analogue for traditional elections. It aims to improve the voting process by formally defining, analyzing and seeking to satisfy difficult and conflicting security properties.

Verifiability aims to assure candidates and voters that all votes have been considered and incorporated into the result. Its close relation with the integrity of the election process and the acceptance of its output, makes verifiability a very important property, extensively studied [8] and implemented in many protocols under computational assumptions or unconditionally [1,16].

Privacy hides the choice of a voter from the talliers, other voters or external agents in order to free her from external pressure and enable her to express her true will. Verifiability without privacy makes no sense. If one assumes that the contents of all votes are publicly known and linked to individuals, then they can in effect be dictated by external agents applying emotional, personal, social and economic pressures. As a result, one cannot be sure that a vote represents the true will of a voter, as the voter could have yielded to these external forces.

⋆ A previous version of this work appeared in the 4th International Joint Conference on Electronic Voting (E-Vote-ID 2019), 1-4 October 2019, Lochau Bregenz, Austria

Thus, the vote cast would not be the one that was intended. In that sense, it would not differ that much from a vote altered by a malicious entity, as is the case with the verifiability threat model.

Privacy has been studied in many variations, in relation to the capabilities of an adversary and its duration. A first level of privacy protections aims to guard against passive adversaries that want to learn the behavior of a particular voter (subset). This has been implemented in two ways: by hiding the contents of the vote or by disassociating the voter identity from the ballot. The former is usually achieved using a threshold cryptosystem with homomorphic properties, while for the latter an anonymity primitive such as mixnets [6] or blind signatures [7] is applied. The actual level of privacy offered depends on the implementation. Homomorphic cryptosystems and mixnets usually provide computational and trust guarantees, as it is generally assumed that there will be an honest subset of participants that will follow the protocol. This means that they will refrain from opening individual votes but will decrypt only the result of the final stage. Blind signatures, on the other hand, can offer information-theoretic protection.

Other stronger types of privacy include *receipt freeness* [3], which protects the voters against 'themselves' and discourages vote selling. *Coercion resistance* [14] concerns active adversaries that aim to dictate voter behavior with methods ranging from abstention, to random voting and impersonation. *Perfect ballot secrecy* [15] proposed in the context of boardroom voting schemes, guarantees that knowledge about the partial tally of a subset of the voters can be computed only by a coalition of all the remaining voters. However, in all of these cases the adversary is computationally restricted.

Everlasting privacy. A less researched variation of privacy is *everlasting privacy.* Its study, formally initiated by Moran and Naor in [19], focuses on preventing vote leaks from attacks by powerful future adversaries. It is motivated by the observation that in most cases, vote privacy is only protected by a cryptosystem the security of which is based on computational assumptions such as the intractability of the Diffie-Hellman problem [4]. These assumptions, however, may be broken or rendered obsolete in the (not too) distant future, as both the theory and the practice of cryptographic attacks always gets better. This means that votes encrypted with small keys are in danger of being revealed, even without the computational assumption being broken. As famously conjectured by Shamir, at the 2006 RSA Conference cryptographers' panel, all cryptographic keys used at that time would remain secure for less than thirty years (cf. [19]).

The situation is made worse, because verifiability requires utilizing public evidence generated by the election system. These pieces of data are meant to be widely available and thus it is easy for an adversary to obtain them, even in part. However, one must bear in mind that the adversaries against voting systems are potentially powerful state agencies with enormous budgets and without time constraints. As a result, they have the capability to collect and store large amounts of election related data. Furthermore, as large-scale elections are organized by the government, these agencies can be considered 'insiders', having access to even private parts of the election transcript. Finally, these agencies

can obtain information exchanged through computer and communication networks, both through mass surveillance as well as with the cooperation of the telecommunication companies.

The problem of privacy is exacerbated, as the information concealed in voting does not lose its value, contrary to protected messages in other common cryptographic scenarios. Indeed, one can easily imagine a future authoritarian regime that tries to gather evidence about its subjects based on past democratic elections in cooperation with the state intelligence agency. This evidence might prompt actions ranging from surveillance to questioning and even more severe repercussions. As noted in [19], such dangers constitute an indirect coercion attempt. In fact, since there are many potential coercers the only rational reaction from a voter fearing all possible adverse scenarios is to abstain. Everlasting privacy seeks to protect the secrecy of individual votes in such scenarios.

Our contribution. In this paper, we propose the first game-based definitions for everlasting privacy. Our definitions are generic, which means that they do not consider the cryptographic primitives that will be used in order to achieve this property. This has not been the case so far (cf. Section 2).

More specifically we consider the adversarial capabilities in terms of both data collection and computational power. To model this, we assume two adversaries: The first is contemporary to the election, where he can participate actively (using corrupted voters) and passively (by monitoring communications between the voters and the authorities). He is computationally bounded, though. The second adversary is computationally unbounded but operates (long) after the election is over. The two adversaries can communicate and as a result the future adversary can obtain election transcripts and auxiliary information collected during the present through means of surveillance by the bounded adversary.

The motivation for this capability stems from the reasonable assumption that there exist powerful entities (e.g. governmental agencies) that might passively hoard election data (among other things as demonstrated by revelations such as Snowden's). It is realistic to assume that a future totalitarian regime will also take control of these agencies (among other things) and have access to their data collection.

By elaborating on the communication options between the present and the future adversary we define two types of everlasting privacy: *strong* and *weak* everlasting privacy, the latter corresponding to the notion of practical everlasting privacy of [2]. Our approach has the added side effect that it associates everlasting privacy with contemporary privacy, which is a relation that, to the best of our knowledge, has not been explored in the literature.

We then apply these threat models of everlasting privacy, against a generic voting scheme. Our analysis focuses particularly on the information gathering capabilities of both adversaries, in relation to the communication channels used. We reason that perfectly hiding commitment schemes do not offer the same levels of protection as anonymous channels, since they cannot hide auxiliary

communication information, that can be utilized by a powerful future adversary with insider information.

2 Related work

The term everlasting privacy was coined in [19]. However, there have been previous works that tackle the same problem, even if they do not use the particular name. For instance, in [9] the voter uses the information theoretically hiding Pedersen commitment scheme to commit to the vote. The openings are then secret shared to the authorities using private channels and homomorphically combined. In order to be verifiable, all exchanged data are stored in a Bulletin Board, modelled as a public broadcast channel with memory. Unfortunately, an adversary that hoards its contents can later use his advanced capabilities to break the privacy of the encrypted shares and reconstruct the votes. Interestingly, in this respect, it can be noted that the blind signature-based protocol of [12], achieves this goal as well, if one assumes a *perfectly* anonymous channel (as Theorem 3 of [12] points). The use of this primitive resembles the shuffling of the ballot box contents, which in traditional elections provides a sense of everlasting privacy to the average voter, who as a human is computationally restricted.

The protocols of Moran and Naor [19,20] further elaborate on providing everlasting privacy through perfectly hiding commitment schemes. They propose a concrete voting system that provides universal verifiability, receipt freeness and everlasting privacy. Additionally, they do not require the voter to perform complex calculations which makes their scheme easily usable by humans. In more details, their proposal consists of two authorities that communicate through a private channel and cooperate in order to produce the commitments that the voter selects. To tally the votes, the authorities work together (privately again) to shuffle the commitments and their openings. The latter are encrypted separately using a homomorphic cryptosystem providing computational secrecy and as a result there are two 'parallel' shuffles. In then end, the perfectly hiding commitments can be safely opened to produce the result. Everlasting privacy is achieved under the assumption that the two authorities do not collude, and the commitment openings are not made public and thus available to the future adversary. If only a single authority is honest, then the scheme of Moran and Naor only provides computational privacy, while if both authorities are corrupted then the system provides only correctness. Despite proving the security of their protocol in the UC framework, the threat model for everlasting privacy isn't formally captured. It merely rests on the perfect secrecy of the commitment scheme and an informal description of the adversary's capabilities. Note that in the future an attacker, that functions as an insider, can have an equivalent effect as if at least one of the authorities was corrupted, which means that the system of [20] does not provide everlasting privacy under this stronger threat model.

Subsequent works further elaborate and generalize this technique of splitting voting data into public and private parts, where the private data are never given to the adversary thus achieving a special version of everlasting privacy - towards

the public. For instance, in [11] the authors apply this procedure to the Helios [1] voting system, by replacing the exponential ElGamal encryptions with Pedersen commitments that are published to the Bulletin Board. Their opening values are sent to the tallier encrypted through private channels. In [10], a relevant primitive - commitment consistent encryption (CCE) is introduced. It allows the voters to derive commitments from their encrypted votes. These commitments are then posted to a public Bulletin Board for verifiability purposes. If they are perfectly hiding, then the voting scheme has everlasting privacy. Tallying takes place in parallel using a private Bulletin Board, where the decryption of the result of the homomorphic combination of the votes takes place. They also provide security definitions for the privacy properties of their particular scheme but not for everlasting privacy in general. Furthermore, in [5] this splitting technique is applied to create two synchronized mixnets that operate in parallel, mixing public commitments and private decommitment values respectively.

The central idea in all the works presented so far is that a future adversary might be more powerful in terms of computing power, but he will lack access to data contemporary to the election or private data available to the authorities. This was noted and formalized in [2] with the notion of *practical* everlasting privacy. However, the formalization used the applied pi-calculus and not the more usual indistinguishability cryptographic games. Using automated tools the authors of [2] proved that the protocols of [20] and [11] possess practical everlasting privacy. However, they did not apply their definition to schemes based on blind signatures and anonymous channels. Moreover, the reliance on private channels assumes an external adversary, an adversary, that is, who has a view of the system similar to the view of the voter. This excludes adversaries that cooperate with the election authorities, who in our opinion are more powerful and more likely to be the perpetrators of a future attack.

More recent works revisit the idea of an anonymous channel as a way to add everlasting privacy to voting schemes. In [17], the voter casts an unencrypted choice to the Bulletin Board along with commitments to their voting credential. The use of an anonymous channel and the fact that the voting credential consists of two parts, prevents a future adversary from associating the choice of a voter with her identity. A variation of this protocol was presented in [18] to offer coercion resistance using deniable vote updating. Along the same lines, in [13], the authors add coercion resistance to the classic protocol of [12]. They also solve the ballot stuffing problem of blind signature based systems using a primitive called Publicly Auditable Blind Signatures, an extension of [21], which forces the election authority to verifiably accept or reject ballots for counting. The advantage of their scheme is that it requires no private channels between voters and authorities as all the election data are found in the Bulletin Board. The blindness of the signatures along with the use of an anonymous channel facilitates everlasting privacy.

3 Voting system syntax

We build our definitions on an abstract election scheme that incorporates ideas from many proposals in the literature in order to be as generic as possible. It is associated with three parameters, the security parameter λ, the number of voters n and the number of possible choices m. The election scheme is controlled by an Election Authority $\mathcal{EA}$, which is stateful and its state is updated in every step of the protocol. In the description that follows we omit this update functionality for simplicity.

We assume the existence of a publicly accessible Bulletin Board where all the election related data is stored. We refer to the current transcript of the Bulletin Board as $\mathcal{BB}$ and we assume that whenever it is used, it contains all the data already written to it. We note that publicly available information such as parameters and public keys are always appended to the public transcript and thus, the $\mathcal{BB}$ would suffice as the public input in the definitions of the scheme. However, we explicitly include such parameters in order to make the algorithms' and protocols' definitions clearer. When we would like to refer to the Bulletin Board as a functionality and not as a data store we use a method invocation-like syntax and we write $\mathcal{BB}()$. Notationally, we use $:=$ for assignment, $=$ for equality, and $\Leftarrow$ for an append operation.

Definition 1. *An election scheme*

$$\mathsf{ES} = (\mathsf{Setup}, \mathsf{Register}, \mathsf{SetupElection}, \mathsf{Authorize}, \mathsf{Vote}, \mathsf{Tally}, \mathsf{Verify})$$

is a tuple of algorithms and protocols executed by the election authority $\mathcal{EA}$, the Bulletin Board $\mathcal{BB}$ and the set of voters $\mathcal{V} = \{\mathcal{V}_1, \ldots, \mathcal{V}_n\}$ parametrized by $\lambda, n, m \in \mathbb{N}$ such that:

- $(\mathsf{params}_{\mathcal{EA}}, sk_{\mathcal{EA}}, pk_{\mathcal{EA}}) := \mathsf{Setup}(1^\lambda)$
  Setup is an algorithm executed by the $\mathcal{EA}$ which on input 1^λ outputs public parameters of the ES and a key pair of the $\mathcal{EA}$ $(\mathsf{sk}_{\mathcal{EA}}, \mathsf{pk}_{\mathcal{EA}})$. The Bulletin Board transcript $\mathcal{BB}$ is appended with $(\mathsf{params}, \mathsf{pk}_{\mathcal{EA}})$.
- $(\mathsf{pk}_i, (\mathsf{sk}_i, \mathsf{pk}_i)) := \mathsf{Register}\langle\mathcal{EA}(\mathsf{sk}_{\mathcal{EA}}), \mathcal{V}_i(), i\rangle$
  $\mathsf{Register}$ is a protocol executed between a voter $\mathcal{V}_i$ and the $\mathcal{EA}$. The common input is the voter id, $i \in \{1, \ldots, n\}$ and the output is a voter public key pk_i (available to both parties) and a secret key sk_i as private output of the voter. The values (i, pk_i) are appended to the $\mathcal{BB}$. We must stress here, that is not obligatory for voters to have a key pair. However, its existence will enable our generic voting scheme to model protocols like [14], that utilize voter credentials for remote voting and coercion resistance.
- $(\mathrm{I}, \mathrm{C}) := \mathsf{SetupElection}(sk_{\mathcal{EA}}, n, m, \mathsf{params}, \text{Election-information})$
  The $\mathcal{EA}$ with input its secret key $\mathsf{sk}_{\mathcal{EA}}$, the number of voters n, the number of choices m and additional election information (e.g. duration) outputs the set of the eligible voters for the election $\mathrm{I} \subseteq \{1, \ldots, n\}$ and the candidate slate C which contains encodings of the choices. The tuple of lists (I, C) is posted to the $\mathcal{BB}$.

- $(\perp, (\mathrm{b}_i, \pi_{\mathrm{b}_i})) := \mathsf{Authorize}\langle \mathcal{EA}(\mathsf{sk}_{\mathcal{EA}}), \mathcal{V}_i(c_i, \mathsf{sk}_i), \mathsf{params}, \mathsf{pk}_{\mathcal{EA}}, \mathsf{pk}_i, \mathrm{I}, \mathrm{C}, \mathcal{BB}\rangle$
  Authorize is a protocol executed between the $\mathcal{EA}$ and a voter $\mathcal{V}_i$. The private input of the $\mathcal{EA}$ is its secret key $\mathsf{sk}_{\mathcal{EA}}$ and the private input of the voter $\mathcal{V}_i$ is her choice of candidate $c_i \in \mathrm{C}$ and her secret key sk_i. The public input consists of the system parameters, the corresponding public keys $\mathsf{pk}_{\mathcal{EA}}, \mathsf{pk}_i$, the set of eligible voters I, the candidate slate C and the contents of the $\mathcal{BB}$. The protocol outputs the ballot b_i, which is a transformation (i.e. encryption) of c_i and a proof π_{b_i} of the correctness of this transformation, usually a Non Interactive Zero Knowledge Proof Of Knowledge. The election authority receives no output from this functionality. We again assume that the protocol transcript is appended to the $\mathcal{BB}$.
- $\mathcal{BB} \Leftarrow \mathsf{Vote}\langle \mathcal{BB}(), \mathcal{V}_i(\mathrm{b}_i, \pi_{\mathrm{b}_i})\rangle$
  Vote is a protocol executed between the voter $\mathcal{V}_i$ and the Bulletin Board $\mathcal{BB}$. The voter $\mathcal{V}_i$ essentially appends the authorized ballot b_i to the election transcript.
- $(\mathbf{T}, \pi_{\mathbf{T}}) := \mathsf{Tally}(\mathsf{sk}_{\mathcal{EA}}, \mathsf{params}, \mathrm{C}, \mathcal{BB})$
  Tally is an algorithm executed by the election authority with input the secret key of the $\mathcal{EA}$, the parameters of the scheme params, the candidate slate C and the transcript $\mathcal{BB}$ of the Bulletin Board and outputs the election tally $\mathbf{T}$ and a proof $\pi_{\mathbf{T}}$. The output is appended to the Bulletin Board $\mathcal{BB}$.
- $\{0, 1\} = \mathsf{Verify}(\mathbf{T}, \mathsf{params}, \mathsf{pk}_{\mathcal{EA}}, \mathcal{BB}, \mathrm{C}, \mathrm{I}, \mathrm{b}_i, \pi_{\mathrm{b}_i}, \pi_{\mathbf{T}})$
  Verify is an algorithm executed by any interested party (voters or public interest organizations) with input the election tally $\mathbf{T}$, the parameters of the scheme params, the public key of the $\mathcal{EA}$ $\mathsf{pk}_{\mathcal{EA}}$, the contents of the Bulletin Board $\mathcal{BB}$, the candidate slate C, the set of eligible voters for the election I, the authorized ballot b and the two proofs $\pi_{\mathbf{T}}, \pi_{\mathrm{b}_i}$. The output is a bit representing the result of the election verification. Verify can indeed be executed by any interested party using all the ballots, for universal verifiability purposes, since all inputs can be found in the $\mathcal{BB}$.

4 Everlasting privacy formalization

We now formally define a voting's system properties regarding privacy. For this reason we consider an adversary, who can corrupt voters and use them with the aim to learn what the honest voters voted. We examine privacy from two aspects: The first concerns 'normal' privacy, which models the protection that voters require during or shortly after the elections. The second applies to the 'everlasting' variation of privacy and models how the voters will be protected (long) after voting has finished. Previously these two definitions were examined independently in the voting literature. However, we note that these properties are intertwined, as an adversary might be motivated to participate in an election, gather evidence by exploiting the voting system and the corrupt voters and possibly use these pieces of information later in time when various constraints might not hold.

More specifically, our adversary is assumed to have the following capabilities:

- He can actively participate in the elections, corrupt voters and collect all data generated by the voting system. During these interventions he is assumed to have computational constraints, as his first goal is to break the privacy of the honest voters during the original election timeframe.
- In the future, he can passively (as there will be no voting taking place) examine the election transcript and extract information about the voters' choices. This adversary is modelled as having unbounded computational capabilities, reflecting the fact that in the future the computational assumptions that protect the votes might not apply due to technological improvements. This future adversary might or might not utilize only the publicly available election information, thus performing either an insider or an outsider attack as discussed in Section 2.

We consider all these cases in our definitions, by assuming a pair of algorithms $(\mathcal{A}, \mathcal{A}')$ where $\mathcal{A}$ is a PPT algorithm and $\mathcal{A}'$ is computationally unbounded. The former participates actively in the election by corrupting voters and the latter looks at the election transcript and (possibly) the information gathered by $\mathcal{A}$ denoted by $view_A$.

Privacy. The privacy game is a variation of the one presented in [16]. We assume that $\mathcal{A}$ is stateful and its state is updated whenever he performs some action in the game. We complete the notation introduced in Section 3 with the use of the symbol $\leftarrow$ to denote the output of an algorithm, and $\leftleftarrows$ for information interchange using a communication channel. Every such communication *leaks* miscellaneous data that are not essential to the protocol but can be used by the adversary to break the system. Such data include network addresses, timestamping information and more. We denote by Aux such miscellaneous data and stress that they will be included in the view of the adversary $view_A$. To denote the execution of one of the functionalities f defined in Section 3 by an entity $\mathcal{E}$ with parameters params we use the following notation: $\mathcal{E}(\mathsf{params}, \mathbf{f})$.

The privacy game is presented in Algorithm 1. It is parameterized with t, the maximum number of voters allowed to be corrupted by the adversary. The challenger $\mathcal{C}$ takes the role of the $\mathcal{EA}$, the $\mathcal{BB}$ and the honest voters. It flips a coin and executes the Setup functionality. After appending its output to the $\mathcal{BB}$ it interacts with each voter in order to complete the $\mathsf{Register}$ protocol. Subsequently $\mathcal{A}$ executes the $\mathsf{SetupElection}$ functionality by providing the challenger with the selection of the eligible voters and the candidates.

The core of the game is the ballot casting phase, which consists of the execution of the $\mathsf{Authorize}$ and Vote functionalities. Before it begins the adversary dynamically decides which voters to corrupt. If voter i is corrupted, then the challenger presents to $\mathcal{A}$ the private key sk_i and gives full control to him. The challenger retains control of the honest voters. The adversary schedules concurrent executions of the $\mathsf{Authorize}$ and Vote functionalities for all voters, in the most *favorable manner* to him. If a voter is corrupted, $\mathcal{A}$ executes these functionalities in her place using a choice $c_{i,\mathcal{A}}$ of his own. If a voter is honest, then $\mathcal{C}$ plays her role, receives 2 selections $c_0, c_1 \in \mathrm{C}$ picked by $\mathcal{A}$ and provides in

Algorithm 1: Privacy Game $\mathsf{Priv}_{\mathcal{A},\Pi,t}(1^\lambda, n, m)$

Input : $1^\lambda, n, m$
Output: $result \in \{0,1\}$

/* Challenger selects random bit and executes setup */
$b \leftarrow \mathcal{C}(\{0,1\})$
$(\mathsf{params}_{\mathcal{EA}}, \mathsf{sk}_{\mathcal{EA}}, \mathsf{pk}_{\mathcal{EA}}) \leftarrow \mathcal{C}(1^\lambda, \textbf{Setup})$
/* Challenger registers voters */
for $i \in [n]$ **do**
  $(\mathsf{sk}_i, \mathsf{pk}_i, \mathsf{Aux}) \leftrightarrows \mathcal{C}(\mathcal{EA}(\mathsf{sk}_{\mathcal{EA}}), \mathcal{V}_i, \textbf{Register})$
end
/* Adversary setups election */
$(\mathrm{I}, \mathrm{C}) \leftarrow \mathcal{A}(\mathsf{params}_{\mathcal{EA}}, n, m, \mathsf{pk}_{\mathcal{EA}}, \{\mathsf{pk}_i\}_{i\in[n]}, \textbf{SetupElection})$
/* Voters perform authorization in the order selected by the adversary */
for $i \in \mathrm{I}$ **do**
  /* Adversary chooses voters to corrupt */
  if $\mathcal{A}(i, \textbf{\textit{corrupt}}) = 1$ **then**
    $V_c \Leftarrow \{i\}$
    /* Adversary performs Authorize for corrupted voters */
    $(\mathrm{b}_i, \pi_{\mathrm{b}_i}, \mathsf{Aux}) \leftrightarrows$
     $\mathcal{A}(\mathcal{EA}(\mathsf{sk}_{\mathcal{EA}}), \mathcal{A}(c_{i,\mathcal{A}}, \mathsf{sk}_i), \mathsf{params}, \mathsf{pk}_{\mathcal{EA}}, \mathsf{pk}_i, \mathrm{I}, \mathrm{C}, \mathcal{BB}, \textbf{Authorize})$
  else
    /* The adversary presents two choices and challenger performs Authorize */
    $(c_0, c_1) \leftarrow \mathcal{A}()$
    $(\mathrm{b}_i, \pi_{\mathrm{b}_i}, \mathsf{Aux}) \leftrightarrows$
     $\mathcal{C}(\mathcal{EA}(\mathsf{sk}_{\mathcal{EA}}), \mathcal{V}_i(c_b, \mathsf{sk}_i), \mathsf{params}, \mathsf{pk}_{\mathcal{EA}}, \mathsf{pk}_i, \mathrm{I}, \mathrm{C}, \mathcal{BB}, \textbf{Authorize})$
  end
end
$V_h := I \backslash V_c$
/* Challenger votes for the set of honest voters in an arbitrary order */
for $i \in V_h$ **do**
  $(\mathcal{BB}, \mathsf{Aux}) \leftrightarrows \mathcal{C}(\mathcal{BB}(), \mathcal{V}_i(\mathrm{b}_i), \textbf{Vote})$
end
/* Adversary votes for the set of corrupted voters. */
for $i \in V_c$ **do**
  $(\mathcal{BB}, \mathsf{Aux}) \leftrightarrows \mathcal{A}(\mathcal{BB}(), \mathcal{A}(\mathrm{b}_i), \textbf{Vote})$
end
/* Tally is executed by the challenger */
$(\mathbf{T}, \pi_{\mathbf{T}}) \leftarrow \mathcal{C}(\mathsf{sk}_{\mathcal{EA}}, \mathsf{params}, \mathrm{C}, \mathcal{BB}, \textbf{Tally})$
/* Define partial tallies for honest voters against c_0, c_1 */
$T_0 := T^{c_0}_{\mathcal{BB}_{V_h}}$
$T_1 := T^{c_1}_{\mathcal{BB}_{V_h}}$
$b' \leftarrow \mathcal{A}(T_0, T_1, \mathsf{params}, \mathcal{BB}, \mathsf{Aux}, \textbf{guess})$
if $T_0 = T_1$ *and* $b = b'$ *and* $|V_c| \leq t$ **then**
  return 1
else
  return 0
end

return the results of $\mathsf{Authorize}, \mathsf{Vote}$ as well as the auxiliary information (such as network traffic, timestamps etc.). The actual choice between c_0, c_1 cast by $\mathcal{C}$ on behalf of the honest voters is defined by a coin flip b. When all voters have finished executions of their protocols, $\mathcal{C}$ executes the Tally functionality and announces the result. $\mathcal{A}$ then receives the full contents of the $\mathcal{BB}$ and auxiliary information Aux and tries to guess the bit b. Note that the adversary has full access to the $\mathcal{BB}$ during the game and as a result he can retrieve its contents at will and not only when he is challenged to guess.

Definition 2. *A voting scheme Π is private if for every PPT algorithm $\mathcal{A}$ there exists a negligible function μ such that for every $n, m \in \mathbb{Z}$ it holds that*

$$\Pr[\mathit{Priv}_{\mathcal{A},\Pi,t}(1^\lambda, n, m) = 1] \leq \frac{1}{2} + \mu(\lambda)$$

Everlasting Privacy. For the everlasting privacy property, we define two games in order to capture the differences in the strategy and knowledge of the future adversary. In both games the adversary $\mathcal{A}'$ is unbounded and invokes the election system that is controlled by the challenger. The difference is that in the $\mathsf{StrongEverPriv}$ game, $\mathcal{A}'$ collaborates with the computationally constrained adversary $\mathcal{A}$, receiving his full state, and utilizes his view including all the auxiliary data he has collected. On the other hand, in the weak everlasting privacy game $\mathcal{A}'$ operates only on the publicly available election data, assumed to be contained in the $\mathcal{BB}$. In both cases $\mathcal{A}'$ tries to guess the result of the coin flip b. Note that the maximum number of possible corruptions t is only required in the $\mathsf{StrongEverPriv}$ game, because of the cooperation of $\mathcal{A}'$ with $\mathcal{A}$. In the weak everlasting privacy game $\mathcal{A}'$ receives only the data of the $\mathcal{BB}$ and has no cooperation with the contemporary adversary. It follows that a voting system cannot have strong everlasting privacy if it has no privacy in the contemporary sense.

Definition 3. *A voting scheme Π has the strong everlasting privacy property if for every pair of algorithms $\mathcal{A}, \mathcal{A}'$, where $\mathcal{A}$ is PPT, there exists a negligible function μ such that for every $n, m \in \mathbb{Z}$ it holds that*

$$\Pr[\mathit{StrongEverPriv}_{\mathcal{A},\mathcal{A}',\Pi,t}(1^\lambda, n, m) = 1] \leq \frac{1}{2} + \mu(\lambda)$$

Definition 4. *A voting scheme Π has the weak everlasting privacy property if for every algorithm $\mathcal{A}'$ there exists a negligible function μ such that for every $n, m \in \mathbb{Z}$ it holds that*

$$\Pr[\mathit{WeakEverPriv}_{\mathcal{A}',\Pi}(1^\lambda, n, m) = 1] \leq \frac{1}{2} + \mu(\lambda)$$

5 Analysis

Having formalized the desired security notions, we now discuss the necessary conditions to satisfy them. In particular, we focus on the data interchanged during the execution of various functionalities.

Algorithm 2: Strong Everlasting Privacy Game $\mathsf{StrongEverPriv}_{\mathcal{A},\mathcal{A}',\Pi,t}(1^\lambda, n, m)$

Input : $1^\lambda, n, m$
Output: $result \in \{0,1\}$

/* Challenger selects random bit */
1 $b \leftarrow_R \{0,1\}$
/* $\mathcal{A}'$ initialises $\mathcal{A}$ with honest voter choices and corruption strategy */
2 $\mathcal{A} \leftarrow \mathcal{A}'(c_0, c_1, V_c)$
/* Challenger executes the election system against $\mathcal{A}$ */
3 $(\mathcal{BB}, \mathsf{Aux}) \leftrightarrows \mathcal{C}(1^\lambda, n, m, c_b, \Pi, \mathcal{A})$
/* tallies for honest voters against c_0, c_1 */
4 $T_0 := T^{c_0}_{\mathcal{BB}_{V_h}}$
5 $T_1 := T^{c_1}_{\mathcal{BB}_{V_h}}$
6 $b' \leftarrow \mathcal{A}'(T_0, T_1, \mathsf{params}, \mathcal{BB}, \mathrm{view}_A, \mathbf{guess})$
7 **if** $T_0 = T_1$ *and* $b = b'$ *and* $|V_c| \leq t$ **then**
8 | return 1
9 **else**
10 | return 0
11 **end**

In Algorithm 1 (line 4) the **Register** functionality generates the voter credentials. We assume that these have private and public parts. All voting systems include a similar functionality, mostly using traditional (i.e. not electronic means). In most cases it does not produce specialized credentials for the voters, except in the case of voting systems based on the JCJ coercion resistance framework [14]. Such systems impose the strictest of requirements for this initial communication between the voter and the authorities, i.e. an untappable channel. The reason for this is that the private data ought to be out of reach for the adversary in case - the coercer - so that the voter can deny a purported private key and successfully apply a coercion resistance strategy. The inconvenience imposed by the untappable channel, is mitigated by the fact that it takes place only once and is later applied to many elections. However, such a channel is not necessary for everlasting privacy.

The more interesting parts of the election system are the execution of the **Authorize** and **Vote** functionalities in Algorithm 1 (lines 13 and 20 respectively). Note that in many systems these functionalities are integrated, as the authorization is assumed to take place 'outside' of the election system, in a manner similar to the registration. In any case, the voter will interact with the election system and post her ballot to the $\mathcal{BB}$ using (a variation of) these functionalities. The output of this process will be the election ballot in encrypted form and auxiliary information (such as network information, timestamps etc.), both of which may be of interest to the future adversary. Its unlimited computational power will

Algorithm 3: Weak Everlasting Privacy Game $\mathsf{WeakEverPriv}_{\mathcal{A}',\Pi}(1^\lambda, n, m)$

Input : $1^\lambda, n, m$
Output: $result \in \{0,1\}$

/* Challenger selects random bit */
$b \leftarrow_R \{0,1\}$
/* Selection of honest voter choices */
$(c_0, c_1) \leftarrow \mathcal{A}'(1^\lambda, n, m)$
/* Challenger executes the election system using c_b */
$(\mathcal{BB}, \mathsf{Aux}) \leftrightarrows \mathcal{C}(1^\lambda, n, m, c_b, \Pi)$
/* tallies for honest voters against c_0, c_1 */
$T_0 := T^{c_0}_{\mathcal{BB}_{V_h}}$
$T_1 := T^{c_1}_{\mathcal{BB}_{V_h}}$
$b' \leftarrow \mathcal{A}'(T_0, T_1, \mathsf{params}, \mathcal{BB}, \mathbf{guess})$
if $T_0 = T_1$ *and* $b = b'$ **then**
  return 1
else
  return 0
end

enable the decryption of the ballot and in turn the linking of the contents of the ballot to the voter.

A system providing everlasting privacy must act on this transfer of information and prevent the data leak. Two options have been proposed on this matter. The first one is to hide the choice of the voter, inside the ballot b_i, using an information theoretically hiding commitment scheme. This means that the decommitment values must be somehow exchanged in order to tally the result. Usually this takes place using private channels with the $\mathcal{EA}$. In our model, this approach does not provide any advantage and is doomed not to possess strong everlasting privacy. This occurs because the strong adversary in Algorithm 2 has access to the private channels and the auxiliary information. More specifically, this approach essentially repeats the posting of the ballot, since an encrypted ballot is essentially the same as a commitment opening, exchanged through a private channel. In both cases the auxiliary information Aux provided to $\mathcal{A}'$ will enable linking the voter to the vote. More specifically in the strong everlasting privacy game (Algorithm 2) the view of $\mathcal{A}$ contains both network identifying information valid during the elections as well as the decommitment values. As a result, $\mathcal{A}'$ can win the game by recovering the votes of the honest voters and guessing the bit. This is not the case with the weak everlasting privacy game (Algorithm 3) as $\mathcal{A}'$ views only the publicly available information in the $\mathcal{BB}$. As a result, he might have access to the vote, but he lacks information about the voter identity.

Another alternative is the use of an anonymous channel during casting. This has the immediate effect that the auxiliary information Aux is in effect nullified,

as the network addresses are hidden. Note that the anonymous channel must not only hide identity information, but the casting order as well. If this is not the case than an adversary that schedules casting to his advantage can break the secrecy of the vote. All he needs to do is have the corrupt voters cast first as shown in Algorithm 1. Subsequently as each honest voter posts her ballot, he can decrypt the last vote cast (using his unlimited computational power) and learn how she voted. There are two ways to thwart this attack: Firstly, there can be an explicit separation of the **Authorize** and **Vote** functionalities. In the beginning, all voters authorize their ballot. After this phase has finished, they cast their votes. This in effect uses the authorization phase to build an anonymity set that hides the order of the votes cast in the voting phase. Alternatively, the same effect can be achieved using an anonymous channel that hides the order of its input messages apart from their source and origin. We consider such an assumption within the range of functionalities provided by such a primitive. Finally, one must note that an anonymous channel can be combined with commitment-based schemes, to nullify the data that leaks during the use of the various communication channels.

One might argue that an unconditionally anonymous channel is required, in order to thwart the information-theoretically powerful future adversary from reversing the anonymity. In our view, however, this is not the case. An anonymous channel might not be in (full) control of the future adversary. It might be distributed, operated (in part) by a non-governmental organization and it might even transcend nation boundaries. As a result, the future totalitarian regime represented by the unbounded adversary, will not have access to the anonymous network in its entirety and subsequently there is no need for it to be based on information theoretically secure primitives. Such a system can successfully succeed in thwarting the adversary from guessing the honest voters' choices as it will not be able to associate them with the real identities and thus achieve both strong and weak everlasting privacy.

6 Conclusion

In this paper we introduced security models for everlasting privacy. Our adversary has the strongest capabilities ever defined in the literature as he is both active during the election by collecting data, as well as in the future where he can break the cryptographic schemes used. Based on this we defined two models of everlasting privacy. Our novel contribution was the modelling of the adversarial capabilities both in terms of computational power and in terms of information context. Using this model, we reasoned that a system based on commitments opened using through channels cannot provide the strongest sense of everlasting privacy, as an adversary with internal knowledge (such as a governmental agency) will have access to both the decommitments and network information. The use of an *independent* anonymous channel, however, will be able to thwart such an attempt. While such a channel is not currently practical, especially at a large scale, our model indicates that research for everlasting privacy will be assisted by its existence, as long as the other properties required by voting systems (e.g. verifiability and election verifiability). Anonymous channels have the

added benefit that they resemble the way traditional elections work and as a result such a system will be more accessible to the voter. Therefore, our paper gives one more reason to continue the research in this direction. In future work, we plan to refine our model and to provide more formal evidence based on concrete instantiations of voting systems and anonymous channels.

Acknowledgements. The authors would like to thank the anonymous reviewers and participants of E-Vote-ID 2019 for their helpful comments and suggestions.

References

1. Adida, B.: Helios: web-based open-audit voting. In: Proceedings of the 17th conference on Security symposium. pp. 335–348. USENIX Association (2008), http://dl.acm.org/citation.cfm?id=1496711.1496734
2. Arapinis, M., Cortier, V., Kremer, S., Ryan, M.: Practical everlasting privacy. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 7796 LNCS, pp. 21–40 (2013). https://doi.org/10.1007/978-3-642-36830-1_2, http://link.springer.com/10.1007/978-3-642-36830-1_2
3. Benaloh, J., Tuinstra, D.: Receipt-free secret-ballot elections (extended abstract). In: Proceedings of the twenty-sixth annual ACM symposium on Theory of computing - STOC '94. pp. 544–553. ACM Press, New York, New York, USA (1994). https://doi.org/10.1145/195058.195407, http://portal.acm.org/citation.cfm?doid=195058.195407
4. Boneh, D.: The Decision Diffie-Hellman problem. In: Buhler, J. (ed.) Algorithmic Number Theory, Third International Symposium, ANTS-III, Portland, Oregon, USA, June 21-25, 1998, Proceedings, Lecture Notes in Computer Science, vol. 1423, pp. 48–63. Springer (2006). https://doi.org/10.1007/bfb0054851, https://doi.org/10.1007/BFb0054851
5. Buchmann, J., Demirel, D., Van De Graaf, J.: Towards a publicly-verifiable mix-net providing everlasting privacy. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 7859 LNCS, pp. 197–204 (2013). https://doi.org/10.1007/978-3-642-39884-1_16, http://link.springer.com/10.1007/978-3-642-39884-1_16
6. Chaum, D.: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Commun. ACM pp. 84–88 (1981)
7. Chaum, D.: Blind Signatures for Untraceable Payments (1982). https://doi.org/10.1007/978-1-4757-0602-4_18, http://link.springer.com/10.1007/978-1-4757-0602-4_18
8. Cortier, V., Galindo, D., Küsters, R., Müller, J., Truderung, T.: SoK: Verifiability Notions for E-Voting Protoc_ols. In: IEEE Security and Privacy Symposium. pp. 779–798 (2016)
9. Cramer, R., Franklin, M., Schoenmakers, B., Yung, M.: Multi-Authority Secret-Ballot Elections with Linear Work. pp. 72–83 (1996). https://doi.org/10.1007/3-540-68339-9_7, http://link.springer.com/10.1007/3-540-68339-9_7
10. Cuvelier, É., Pereira, O., Peters, T.: Election verifiability or ballot privacy: Do we need to choose? In: Lecture Notes in Computer Science (including subseries

Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 8134 LNCS, pp. 481–498 (2013). https://doi.org/10.1007/978-3-642-40203-6_27, http://link.springer.com/10.1007/978-3-642-40203-6_27

11. Demirel, D., Graaf, J.V.D., Araújo, R.: Improving Helios with Everlasting Privacy Towards the Public. EVT/WOTE'12 Proceedings of the 2012 international conference on Electronic Voting Technology/Workshop on Trustworthy Elections (2012)
12. Fujioka, A., Okamoto, T., Ohta, K.: A practical secret voting scheme for large scale elections (1992). https://doi.org/10.1007/3-540-57220-1_66, http://link.springer.com/10.1007/3-540-57220-1_66https://doi.org/10.1007/3-540-57220-1_66
13. Grontas, P., Pagourtzis, A., Zacharakis, A., Zhang, B.: Towards everlasting privacy and efficient coercion resistance in remote electronic voting. In: Zohar, A., Eyal, I., Teague, V., Clark, J., Bracciali, A., Sala, F., Massimiliano, P. (eds.) Financial Cryptography and Data Security (FC 2018). Lecture Notes in Computer Science, vol 10958, pp. 210–231. Springer (2019). https://doi.org/10.1007/978-3-662-58820-8_15, https://eprint.iacr.org/2018/215.pdf
14. Juels, A., Catalano, D., Jakobsson, M.: Coercion-resistant electronic elections. In: Atluri, V., di Vimercati, S.D.C., Dingledine, R. (eds.) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 6000 LNCS, pp. 37–63. ACM (2005). https://doi.org/10.1007/978-3-642-12980-3_2, http://doi.acm.org/10.1145/1102199.1102213
15. Kiayias, A., Yung, M.: Self-tallying elections and perfect ballot secrecy. In: Naccache, D., Paillier, P. (eds.) Public Key Cryptography, 5th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2002, Paris, France, February 12-14, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2274, pp. 141–158. Springer (2002). https://doi.org/10.1007/3-540-45664-3_10, https://doi.org/10.1007/3-540-45664-3_10
16. Kiayias, A., Zacharias, T., Zhang, B.: End-to-end verifiable elections in the standard model. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 9057, pp. 468–498 (2015). https://doi.org/10.1007/978-3-662-46803-6_16, http://link.springer.com/10.1007/978-3-662-46803-6_16
17. Locher, P., Haenni, R.: Verifiable internet elections with everlasting privacy and minimal trust. In: Haenni, R., Koenig, R.E., Wikström, D. (eds.) E-Voting and Identity - 5th International Conference, VoteID 2015, Bern, Switzerland, September 2-4, 2015, Proceedings. Lecture Notes in Computer Science, vol. 9269, pp. 74–91. Springer (2015). https://doi.org/10.1007/978-3-319-22270-7_5, https://doi.org/10.1007/978-3-319-22270-7_5
18. Locher, P., Haenni, R., Koenig, R.E.: Coercion-resistant internet voting with everlasting privacy. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 9604 LNCS, pp. 161–175 (2016). https://doi.org/10.1007/978-3-662-53357-4_11, http://link.springer.com/10.1007/978-3-662-53357-4_11
19. Moran, T., Naor, M.: Receipt-Free Universally-Verifiable Voting with Everlasting Privacy. pp. 373–392 (2006). https://doi.org/10.1007/11818175_22, http://link.springer.com/10.1007/11818175_22
20. Moran, T., Naor, M.: Split-ballot voting. ACM Transactions on Information and System Security **13**(2), 1–43 (feb 2010). https://doi.org/10.1145/1698750.1698756, http://portal.acm.org/citation.cfm?doid=1698750.1698756

21. Zacharakis, A., Grontas, P., Pagourtzis, A.: Conditional blind signatures. In: 7th International Conference on Algebraic Informatics (short version) (2017), full version available on: http://eprint.iacr.org/2017/682