Relations Among Privacy Notions

Jens-Matthias Bohli and Andreas Pashalidis

NEC Laboratories Europe
Kurfürsten-Anlage 36
69115 Heidelberg, Germany

Abstract. This paper presents a hierarchy of privacy notions that covers multiple anonymity and unlinkability variants. The underlying definitions, which are based on the idea of indistinguishability between two worlds, provide new insights into the relation between, and the fundamental structure of, different privacy notions. We apply the definitions to a number of privacy-preserving systems, namely group signatures, voting systems, and anonymous communication systems, and show how they relate to existing definitions.

1 Introduction

With the growing number of services and information offered in the digital world, the number of situations where there is a need to hide the correspondence between digital elements and the people that cause their appearance, is also increasing. A variety of privacy protecting systems address this need; anonymous communication systems, for example, hide how transmitted messages correspond to their senders (and their recipients); group signatures hide the identity of the signer of a given message, and secret voting schemes hide the identity of the voter who cast any given ballot. In general, a system is said to 'provide privacy' if it hides, perhaps to an extent, the correspondence between the elements it outputs, and its users.

What *exactly* it means for any given privacy protecting system to provide privacy naturally varies between system types. The privacy definition for group signatures [5], for example, differs from the one for anonymous credentials [8]. Similarly, privacy for voting schemes [1] is defined differently from privacy in the setting of anonymous communication [24]. Despite efforts for a consistent terminology [32], formal treatments *seem* to define privacy in an inconsistent and sometimes even contradictory manner; while, for example, some authors assert that 'anonymity and unlinkability are technically the same property' [5], others show that, although related, they are, in fact, distinct [24], and others insist that they are independent [25]. Unfortunately, it is not only the terminology that is used inconsistently; due to the discrepancies between the formal models, the resulting privacy notions themselves turn out to be incomparable. It remains unclear whether or not it is possible to construct a *single* formal framework in which privacy notions pertaining to *different* system types can be defined in a consistent and comparable manner.

Our Contributions: This work can be seen as first step towards a formal framework that aims to define multiple privacy notions in an application-agnostic manner. By so doing, it provides new insights into the inner structure of privacy notions. Starting from a generic system model that potentially hides the correspondence between digital elements and the users that cause their appearance, we systematically analyse different degrees to which this correspondence may be hidden, and place the resulting privacy notions into a well-characterised hierarchy. Furthermore, we examine the class of 'stateless' and 'online' systems, and show why only some privacy notions apply to these classes. Finally, we place existing definitions for group signatures, anonymous communication, and secret voting in the context of our framework. This enables us, on the one hand, to understand the relationship between, and to compare, these traditionally disconnected privacy notions. On the other hand, it highlights a largely unexplored space of theoretically possible notions some of which may be of practical interest.

Related Work: The framework introduced in [24] has certain commonalities with the framework introduced in this paper; both define, for example, a hierarchy of privacy notions based on the principle that an adversary may break any privacy notion *except* the one of interest, and both follow the idea of left-or-right security introduced in [4]. However, the framework in [24] appears to be specific to anonymous communication systems. Moreover, the hierarchy of privacy notions defined in this paper is richer; when mapped to anonymous communication systems, notions beyond those considered in [24] arise. The framework in [25] also has certain commonalities with ours; both support, for example, the specification of privacy notions against adversaries with partial knowledge about a function. However, in contrast to the framework introduced in this paper, the one in [25] does not consider probabilistic adversaries, and is only applied to anonymous communication systems. Moreover, it is unclear how its privacy definitions map to existing and established application-specific ones.

Other related work includes the literature on *measuring* privacy (e.g. [3, 10–16, 19, 23, 29, 34, 35, 37]). The proposed metrics appear, however, to pertain to particular privacy notions, if not system types. Multiple, sometimes inconsistent metrics for the same notion have also been proposed. While, for example, the metric in [15], proposed for the anonymity in the setting of anonymous communication systems, focuses on the relationship between incoming and outgoing messages, the metric proposed in [21] focuses on the relationship between senders and receivers. Similarly, the metric for unlinkability proposed in [19, 36] does not take into account the skewness of the adversary's view on possible solutions, while the metric proposed in [18] does. The only work we are aware of that places multiple privacy notions into a single framework [31], does not relate the metrics it defines to privacy definitions from the cryptographic literature. It is important to note that most privacy metrics cited above are probabilistic. That is, they measure degree to which a system provides privacy. In contrast to this, the privacy definitions in this paper are 'all-or-nothing'; a system either provides or does not provide a given privacy notion.

While the most popular adversarial model in the anonymous communication literature is perhaps that of the 'global passive' adversary (see, for example, [26, 30, 33]), in this paper we consider an adaptive adversary that may corrupt users. This is in line with definitions from group signatures [5], anonymous credentials [8], and some of the literature on anonymous communication (e.g. [6, 16]). Moreover, our privacy definitions classify systems as either succeeding, or failing to provide a given privacy notion; while this is in contrast with some works on anonymous communication that consider 'soft', probabilistic measures (see, for example, [16, 23]), it, too, is in line with works on group signatures and some of the literature on anonymous communication systems (see, for example, [21]).

Outline: The rest of this paper is organised as follows. The next section introduces our notation and formal model, and Section 3 presents the hierarchy of privacy notions and examines its structure. Sections 3.2 and 3.3 examine 'stateless' and 'online' systems and show why only some privacy notions apply in these types of system. Section 4 examines group signatures, anonymous communication, and secret voting systems in the context of the hierarchy. Section 5 concludes.

2 Preliminaries

This section introduces our notation and formal model. In particular, the next section introduces the class of systems that are considered in this paper, Section 2.2 introduces the different privacy notions considered, and Section 2.3 describes the adversarial model.

2.1 System model

In this paper, we consider systems that may be *sequentially invoked* a finite number of times and that, for each invocation, produce an element $e \in \{0,1\}^*$. It is required that each invocation is uniquely associated with a user and with an input parameter $\alpha \in A$, where A is a system-specific

parameter space, that may influence the behaviour of the system. It is furthermore required that each user is identified by means of a unique identifier from an identifier space; we use $\mathbb{N}$ for this purpose, but any large enough space can be used without loss of generality.

We assume that the system, denoted by Φ^A in the sequel, produces its output in batches of potentially varying sizes. That is, it is assumed that, on input a batch of invocations $(u_1, \alpha_1), (u_2, \alpha_2), \ldots, (u_c, \alpha_c) \in (\mathbb{N} \times A)^c$, Φ^A outputs a sequence $((e_1, \ldots, e_c), \beta)$, where the sequence $(e_1, \ldots, e_c)$ contains the elements that Φ^A produced as a result of the invocations. The order in which the elements appear in this sequence is determined by the system, and may differ from the order of the invocations. In particular, e_i is the element that Φ^A produces for the invocation $(u_{\pi(i)}, \alpha_{\pi(i)})$, for some potentially secret Φ^A-specific permutation π. Finally, $\beta \in \{0,1\}^*$ denotes some additional information that Φ^A outputs and that pertains to a batch as a whole, i.e. that is not associated with any specific invocation.

Remark 1. The system output being generated in batches models the behaviour of certain privacy-protecting systems that do not generate an output immediately after each invocation, but rather collect several inputs before producing some output. Mix networks [9] and secret voting schemes [1], for example, operate in this way: mix networks can provide privacy only if they forward multiple messages at a time, and secret voting schemes require multiple votes for different candidates to be cast before the tally is published in order to provide privacy. However, some privacy protecting systems, for example group signatures [5], do not exhibit this behaviour, i.e. have batch size 1. These systems are examined in Section 3.3.

2.2 Privacy model

Let x denote the number of times the system is invoked during its lifetime. The correspondence between (the serial numbers of) the elements that occur during the lifetime of a system Φ^A and the set of its users is modelled as a function $f \in \mathfrak{F}$, where $\mathfrak{F} = \{f : \{1, 2, \ldots, x\} \to \mathbb{N}\}$ is the space of functions that map the serial number of each output element to the (identifier of the) user it corresponds to. The privacy notions considered in this paper describe potentially different degrees to which f remains hidden from an adversary. The adversary's goal is to identify f, or some 'interesting property' of f, possibly with respect to some subset of elements, through interaction with, or observation of, Φ^A. We consider the following properties of f with respect to a subset $I \subseteq \{1, 2, \ldots, x\}$ of element serial numbers, which may be of interest to an adversary.

$U_{f,I} = \{f(i) : i \in I\} \subset \mathbb{N}$ denotes the *participant set*, i.e. the set of user identifiers that are associated with the elements in I.

$Q_{f,I} = \{(u, \#u_{f,I}) : u \in U_{f,I}\}$, where $\#u_{f,I} = |\{i \in I : f(i) = u\}| \in \{1, 2, \ldots, |U_{f,I}|\}$, denotes *usage frequency set*, i.e. the collection of records that indicate how many elements correspond to each participant from I's participant set.

$P_{f,I} = \{I'_1, I'_2, \ldots, I'_{|U_{f,I}|}\} \vdash I$ denotes the *linking relation*, i.e. the partition of I that is induced by f. That is, $P_{f,I}$ denotes the partition that divides I into non-overlapping subsets such that, for all $i, i' \in I'_j$, $f(i) = f(i')$. Note that $\bigcup_j I'_j = I$.

In the sequel, omission of the modifier I implies that the property under consideration refers to the entire lifetime of the system, i.e. that $I = \{1, 2, \ldots, x\}$. Given the above properties, and based on the principle that the adversary should be allowed to break any privacy notion *except* the one of interest, we derive the following privacy notions. These notions are further formalised in Section 3.

- Strong anonymity, denoted SA: A system that provides SA does not enable the adversary to learn any information about how elements correspond to users, i.e., it does not leak any information about f.

– Participation hiding, denoted PH: A system that provides PH does not leak any information about f beyond the number of participants $|U_f|$. In particular, it does not enable the adversary to learn any information about the participant set U_f beyond its size.
– Strong unlinkability, denoted SU: A system that provides SU does not leak any information about f beyond the participant set U_f. In particular, it does not enable the adversary to link, or to unlink, different elements beyond the extent it can do so based on knowledge of U_f. In other words, a system that provides SU does not leak any information about the linking relation P_f beyond what is leaked by U_f.
– Weak unlinkability, denoted WU: A system that provides WU does not leak any information about f beyond the usage frequency set Q_f. In particular, it does not enable the adversary to link, or to unlink, different elements beyond the extent it can do so based on knowledge of Q_f. In other words, a system that provides WU does not leak any information about the linking relation P_f beyond what is leaked by Q_f.
– Pseudonymity, denoted PS: A system that provides PS does not leak any information about f beyond the linking relation P_f. In particular, it does not enable the adversary to learn any information about the participant set U_f beyond what it learns from P_f. This notion is called 'pseudonymity' because each equivalence class in P_f (which is assumed to be known) can be given a unique label, or 'pseudonym'.
– Anonymity, denoted AN: A system that provides AN does not leak any information about f beyond the linking relation P_f and the participation set U_f. Intuitively, a system that provides AN may enable the adversary divide all elements into non-overlapping groups, and also determine the set of participants they correspond to, but does not enable it to determine which group corresponds to which participant.
– Weak anonymity, denoted WA: A system that provides WA does not leak any information about f beyond the linking relation P_f and the usage frequency set Q_f. Similarly to AN, WA requires that the system hides the correspondence between element groups and participants. However, since knowledge of Q_f may enable the adversary to at least partially establish this correspondence, systems that provide WA (but not AN) hide less information about it that systems that provide AN (which do not reveal any information about it).

2.3 Adversarial model

This section specifies the adversarial model considered in this paper. The adversary, denoted by $\mathcal{A}$ in the sequel, adaptively controls the usage of Φ^A, and is allowed to corrupt users, i.e. to obtain a copy of their private information and their internal state. Its interaction with Φ^A is modelled by means of an experiment that a challenger arranges for $\mathcal{A}$. At the beginning of this experiment, the user identifier space $\mathbb{N}$ and, if necessary, a security parameter $k \in \mathbb{N}$, are fixed and Φ^A is set up. The experiment, depicted in Figure 1, starts with the challenger selecting a bit $b \in \{0,1\}$ uniformly at random, and by setting the initial value of the input counter c to zero. The challenger then offers the following interfaces to $\mathcal{A}$, through which the system can be controlled.

– `input`$((\cdot,\cdot),(\cdot,\cdot))$: on input $((u_0,\alpha_0),(u_1,\alpha_1)) \in (\mathbb{N} \times A)^2$, the challenger first increases the counter c by one and then remembers (u_b,α_b) as $(u_\mathsf{c},\alpha_\mathsf{c})$.
– `nextBatch()`: on reception of this query type, the challenger invokes Φ^A on input the 'remembered' values $(u_1,\alpha_1),(u_2,\alpha_2),\ldots,(u_\mathsf{c},\alpha_\mathsf{c})$ and outputs the system's output. We say that the challenger outputs a batch of size c in this case.[1] Subsequently, the input counter c is reset to zero.

[1] The specification of β, π, and how α it influences the output of Φ^A, is specific to Φ^A. Moreover, if the adversary is polynomially bounded, then the length of α_0,α_1,β and all e_i must be polynomial in the system's security parameter.

– `corrupt(·)`: on input $u \in \mathbb{N}$, the challenger outputs the internal state of the user identified by u. The specification of the information that is returned to $\mathcal{A}$ is specific to Φ^A.

$\mathcal{A}$ may issue a number of queries over these interfaces and, at some point in time, outputs a guess bit $g \in \{0,1\}$. We say that $\mathcal{A}$ wins the experiment if and only if $g = b$, and its advantage is given by $\mathbf{Adv}_{\Phi^A,\mathcal{A}}(k) = \Pr(\mathbf{Exp}^{\text{priv}-0}_{\Phi^A,\mathcal{A}}(k)) - \Pr(\mathbf{Exp}^{\text{priv}-1}_{\Phi^A,\mathcal{A}}(k))$.

Experiment $\mathbf{Exp}^{\text{priv}-\text{b}}_{\Phi^A,\mathcal{A}}(k)$
$b \leftarrow \{0,1\}$;
$g \leftarrow \mathcal{A}^{\texttt{input}((\cdot,\cdot),(\cdot,\cdot)),\texttt{nextBatch}(),\texttt{corrupt}(\cdot)}$
return $g == b$

Fig. 1. Experiment $\mathbf{Exp}^{\text{priv}-\text{b}}_{\Phi^A,\mathcal{A}}(k)$.

Some notation is in order. Let κ denote the number of `nextBatch` queries $\mathcal{A}$ has issued up to the point in time it outputs g in an $\mathbf{Exp}^{\text{priv}-\text{b}}_{\Phi^A,\mathcal{A}}(k)$ experiment. For all $1 \leq j \leq \kappa$, let c_j denote the size of the batch that the challenger output as a result of $\mathcal{A}$'s jth `nextBatch` query, and let π_j denote the permutation applied by Φ^A for the jth batch. Furthermore, let $x = \sum_{j=1}^{\kappa} c_j$ denote the total number of $\texttt{input}((\cdot,\cdot),(\cdot,\cdot))$ queries. For all $1 \leq i \leq x$, we denote by $u_{0,i}$ (resp. $u_{1,i}$, $\alpha_{0,i}$, $\alpha_{1,i}$) the value of u_0 (resp. u_1, α_0, α_1) in $\mathcal{A}$'s ith $\texttt{input}((u_0,\alpha_0),(u_1,\alpha_1))$ query. We further define the subsets of invocation serial numbers $I_1 = \{1,2,\ldots,c_1\}, I_2 = \{c_1+1, c_1+2, \ldots, c_1+c_2\}, \ldots, I_\kappa = \{c_1+c_2+\cdots+c_{\kappa-1}+1, c_1+c_2+\cdots+c_{\kappa-1}+2, \ldots, x\}$, and the 'global inverse permutation' Π as the permutation that maps the serial number of all elements that are output during the experiment to the serial number of their corresponding invocation. That is, Π permutes $(1,2,\ldots,x)$ such that, for all $1 \leq i \leq x$, $\Pi(i) = \pi_j^{-1}(i - \sum_{j'=1}^{j-1} c_{j'}) + \sum_{j'=1}^{j-1} c_{j'}$, where $j \in \{1,2,\ldots,\kappa\}$ is such that $i \in I_j$. Finally, the functions f_0, f_1 are defined such that, for all $i \in \{1,2,\ldots,x\}$, $f_0(i) = u_{0,\Pi(i)}$ and $f_1(i) = u_{1,\Pi(i)}$.

3 Hierarchy of privacy notions

This section formalises the privacy notions introduced in Section 2.2 and shows how they relate to each other. We begin by defining the following seven notions of function distinguishability.

Definition 1. *Two functions $f, f' \in \mathfrak{F}$, $f \neq f'$, are said, with respect to a subset of invocations $I \subseteq \{1,2,\ldots,x\}$, to be*

SA-distinguishable *in any case,*
PH-distinguishable *if and only if $|U_{f,I}| = |U_{f',I}|$,*
SU-distinguishable *if and only if $U_{f,I} = U_{f',I}$,*
WU-distinguishable *if and only if $Q_{f,I} = Q_{f',I}$,*
PS-distinguishable *if and only if $P_{f,I} = P_{f',I}$,*
AN-distinguishable *if and only if $P_{f,I} = P_{f',I}$ and $U_{f,I} = U_{f',I}$, and*
WA-distinguishable *if and only if $P_{f,I} = P_{f',I}$ and $Q_{f,I} = Q_{f',I}$.*

We are now ready to present our main privacy definitions.

Definition 2. *A privacy protecting system Φ^A is said to unconditionally (resp. statistically) provide privacy notion X^* for some $X \in \{\mathsf{SA}, \mathsf{PH}, \mathsf{SU}, \mathsf{WU}, \mathsf{PS}, \mathsf{AN}, \mathsf{WA}\}$ if and only if f_0 and f_1 are X-distinguishable with respect to all $I \in 2^{\{I_1,\ldots,I_\kappa\}}$, and, for all $\mathcal{A}$, $\mathbf{Adv}_{\Phi^A,\mathcal{A}}(k) = 0$ (resp. $\mathbf{Adv}_{\Phi^A,\mathcal{A}}(k) \leq \epsilon(k)$ for some negligible function ϵ). Moreover, Φ^A is said to computationally provide privacy notion X^* if and only if it statistically provides X^* and the running time of $\mathcal{A}$ is polynomial in k.*

The above privacy notions are very strong because they require that $\mathcal{A}$ does not obtain any advantage neither by corrupting users, nor on the basis of the parameter values that it passes in its `input` queries. We therefore require weaker notions that take corrupted users into account and that limit $\mathcal{A}$'s ability to distinguish between the two worlds on the basis of parameter values. The following notion of function indistinguishability is therefore necessary.

Definition 3. *Two functions $f, f' \in \mathfrak{F}$ are said to be* indistinguishable *with respect to a subset of (corrupted) users $\hat{U} \subseteq \mathbb{N}$, denoted by $f \approx_{\hat{U}} f'$, if and only if $\{i : i \in \{1, 2, \ldots, x\}, f(i) \in \hat{U}\} = \{i : i \in \{1, 2, \ldots, x\}, f'(i) \in \hat{U}\}$, i.e. if and only if the pre-image of $\hat{U}$ is identical in f and f'.*

Let $\hat{U} \subseteq \mathbb{N}$ denote the set of users that $\mathcal{A}$ has corrupted up to the point in time it outputs g, and let $A_0 = (\alpha_{0,1}, \alpha_{0,2}, \ldots, \alpha_{0,x})$ and $A_1 = (\alpha_{1,1}, \alpha_{1,2}, \ldots, \alpha_{1,x})$ denote the parameter sequences in the two worlds. We now present our weaker, more realistic notions.

Definition 4. *A privacy protecting system Φ^A is said to unconditionally (resp. statistically, computationally) provide privacy notions X°, X^+ and X for some $X \in \{\mathsf{SA}, \mathsf{PH}, \mathsf{SU}, \mathsf{WU}, \mathsf{PS}, \mathsf{AN}, \mathsf{WA}\}$, if and only if it provides X^* and $\mathcal{A}$ is restricted as shown below.*

Privacy notion	*Restrictions*
X°	$A_0 = A_1$
X^+	$f_0 \approx_{\hat{U}} f_1$
X	$A_0 = A_1$ *and* $f_0 \approx_{\hat{U}} f_1$

3.1 Relations between notions

For all $X \in \{\mathsf{SA}, \mathsf{PH}, \mathsf{SU}, \mathsf{WU}, \mathsf{PS}, \mathsf{AN}, \mathsf{WA}\}$ it trivially holds that $X^* \Rightarrow X^+ \Rightarrow X$ and $X^* \Rightarrow X^\circ \Rightarrow X$ because $\mathcal{A}$ is more restricted in its choices in the context of X than it is in the contexts of X^+ and X°, and more restricted in the contexts of X^+ and X° than it is in the context of X^*.

The 'plain' privacy notions X are, perhaps, the most typical ones as they are concerned with the amount and type of information the system leaks *exclusively* on the basis of the identities of honest users. The notions X^+ are stronger, in the sense that a system providing some notion X^+ must not enable $\mathcal{A}$ to distinguish between system invocations on the basis of the parameters passed to the system; the system must ensure that the output corresponding to different users is indistinguishable, irrespective of the two users' potentially different input.

The privacy notions X° can be seen as a form of 'forward/backward privacy', analogous to notions of forward and backward security for encryption schemes. *Forward privacy* means that, even if a user is compromised via a `corrupt` query, the user's system interactions that occurred prior to this corruption remain private. Similarly, *backward privacy* means that system interactions of a user remain private, even if the user was corrupted prior to these interactions. Section 4.1 shows that the established privacy notion for group signatures a forward/backward privacy notion.

The privacy notions X^* are very strong, in the sense that a system providing X^* protects the privacy of *all* users, honest and corrupted alike, *and* does not enable $\mathcal{A}$ to distinguish between system invocations on the basis of the parameters passed to the system. That is, a system provides the notion X^* only if it provides X^+ and X° at the same time.

Figure 2 shows further relations between different privacy notions. These relations follow from the facts that knowledge of Q_f implies knowledge of U_f, and that knowledge of U_f or P_f implies knowledge of $|U_f|$. The same hierarchy also applies to the privacy notions X^+, X°, and X^*.

Remark 2. Intuitively, 'unobservability' is a privacy notion that ensures that $\mathcal{A}$ cannot determine whether or not a system invocation takes place. A system can only provide unobservability if it supports the notion of a 'void' invocation. That is, potentially unobservable systems must accept 'normal' invocations, i.e. invocations that are associated with some user/parameter pair from $\mathbb{N} \times A$,

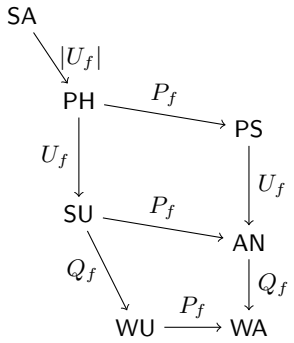

Fig. 2. Relations between privacy notions. The arrow labels indicate the property about f that the system may reveal. From left to right, more information about the linking relation P_f is revealed; from top to bottom, more information about the user involvement Q_f is revealed.

and void invocations, i.e. invocations that are not associated with anything. A system can only be unobservable if it produces an element for void invocations that is indistinguishable from the elements it produces as a result of normal invocations. Since our system model described in Section 2.1 does not support systems that that accept void invocations, our framework does not include an 'unobservability' privacy notion. However, this discussion demonstrates that extending the framework in this direction is straight-forward.

3.2 Stateless systems

This section examines the privacy notions that apply to a particular class of privacy protecting systems which we call 'stateless'. Intuitively, stateless systems are systems whose outputs are encrypted, and that permute their outputs uniformly. Consider an experiment $\mathbf{Exp}^{\text{priv}-\text{b}}_{\Phi^A,\mathcal{A}}(k)$. A *parameter-based element distinguisher* is an algorithm that, on input any element e that is output during the experiment, and the parameter sequences A_0 and A_1, outputs a guess g for b.

Definition 5. *A system Φ^A is said to be unconditionally (resp. statistically)* stateless *if and only if, for all batches,*

- *π is chosen uniformly at random,*
- *neither the elements it outputs, nor β contains any information about the chosen permutation π (or, equivalently, about the order in which the invocations of that batch were made), and*
- *no parameter-based element distinguisher has a positive (resp. non-negligible) advantage over random guessing.*

Moreover, Φ^A is said to be computationally stateless if the first two conditions hold and no polynomial time parameter-based distinguisher has a non-negligible advantage over random guessing.

In other words, the order in which the elements are produced by a stateless system is independent from the order of the invocations in the respective batch, and parameter values do not enable $\mathcal{A}$ to obtain an advantage in distinguishing between the two worlds.

Lemma 1. *If Φ^A is unconditionally stateless, and if $Q_{f_0,I} = Q_{f_1,I}$ for all $I \in \{I_1, \ldots, I_\kappa\}$, then challenger's output in an $\mathbf{Exp}^{\text{priv}-\text{b}}_{\Phi^A,\mathcal{A}}(k)$ experiment is independent from b.*

Proof. Assume that the challenger's output depends on b for some I. Since $Q_{f_0,I} = Q_{f_1,I}$, for each batch element in the $\mathbf{Exp}^{\mathrm{priv}-0}_{\Phi^A,\mathcal{A}}(k)$ experiment, there exists a batch element in the $\mathbf{Exp}^{\mathrm{priv}-1}_{\Phi^A,\mathcal{A}}(k)$ experiment, such that the user/parameter pair associated with the corresponding invocations of these elements are identical. Since, due to the assumed statelessness, each element depends *only* on this pair, the *order* in which batch elements appear in the $\mathbf{Exp}^{\mathrm{priv}-\mathrm{b}}_{\Phi^A,\mathcal{A}}(k)$ experiment must contain some information about b. This, however, contradicts the fact that π is selected uniformly at random, and the result follows. □

Lemma 1 has the following interesting implication.

Corollary 1 *All unconditionally (resp. statistically, computationally) stateless systems unconditionally (resp. statistically, computationally) provide* WU *and* WA*.*

The resulting hierarchy of privacy notions, sketched in Figure 3 does therefore not include WU and WA.

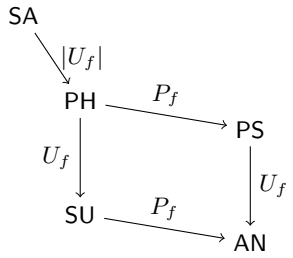

Fig. 3. Relations between privacy notions for stateless systems.

3.3 Online systems

This section examines systems that process every input individually, i.e. systems that have a constant batch size equal to one. While such systems, which we call 'online' systems, enable $\mathcal{A}$ to trivially keep track of the mapping of `input` queries and the elements produced by the system, our definition still requires $\mathcal{A}$ to determine whether it is interacting in the left or the right world. Nevertheless, the mere fact that $\mathcal{A}$ can unambiguously determine which output elements correspond to which invocation serial numbers, has implications to the introduced hierarchy of privacy notions.

Lemma 2. *Consider two functions $f, f' \in \mathfrak{F}$. If $U_{f,I} = U_{f',I}$ for all $I \in \big\{\{1\},\{2\},\ldots,\{x\}\big\}$, then $f = f'$.*

Proof. Assume that $f \neq f'$, i.e. that there exists at least one $i \in \{1,2,\ldots,x\}$ such that $f(i) \neq f'(i)$. Then $U_{f,\{i\}} \neq U_{f',\{i\}}$, contradicting the assumption. □

The implication of Lemma 2 is that, for online systems, there exist no functions f_0 and f_1, $f_0 \neq f_1$, that are X-distinguishable for any $X \in \{\mathsf{SU},\mathsf{WU},\mathsf{AN},\mathsf{WA}\}$. Moreover, two functions that are PH-distinguishable are also PS-distinguishable. This can be seen easily, since, for all $I \in \big\{\{1\},\{2\},\ldots,\{x\}\big\}$, $|U_{f,I}| = |U_{f',I}| = 1$, and both P_{f_0} and P_{f_1} divide $\{1,2,\ldots,x\}$ into partitions of singletons. Hence, $P_{f_0} = P_{f_1}$. The resulting collapsed hierarchy of privacy notions is sketched in Figure 4.

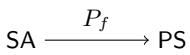

Fig. 4. Privacy notions for online-systems.

4 Applications

This section places the privacy definitions concerning group signatures, voting systems, and anonymous communication systems into the hierarchy introduced in the previous section.

4.1 Group signatures

Group signatures represent an important class of privacy protecting system. A group signature system consists of four algorithms $(\mathsf{GKg}, \mathsf{GSig}, \mathsf{GVf}, \mathsf{Open})$, as follows [5].

- The randomised *group key generation algorithm* GKg takes as input a security parameter $k \in \mathbb{N}$, and returns a tuple $(gpk, gmsk, gsk)$, where gpk is the *group public key*, $gmsk$ is the *group manager's secret key*, and gsk is an n-vector of keys where $gsk[u]$ is the *secret signing key* of user identified by $u \in \mathbb{N}$, and where $n \in \mathbb{N}$ is polynomially bounded in k.
- The randomised *group signing algorithm* GSig takes as input a secret signing key $gsk[u]$ and a message $m \in \mathcal{M}$, where $\mathcal{M}$ is the system's message space, to return a signature of m under $gsk[u](u \in \mathbb{N})$.
- The deterministic *group signature verification algorithm* GVf takes as input the group public key gpk, a message m, and a candidate signature σ for m to return either 1 or 0.
- The deterministic *opening algorithm* Open takes as input the group manager secret key $gmsk$, a message m, and a signature σ of m to return an identifier $u \in \mathbb{N}$ or the symbol $\perp$ to indicate failure.

Translated to the system model of Section 2.1, the parameter space of a group signature scheme is the its message space. That is, $A_{\mathtt{gs}} = \mathcal{M}$. Since users compute and independently release signatures by themselves, the adversary is able to observe isolated system invocations. Thus, group signature schemes are online systems, and, hence, the only applicable privacy notions are SA and PS. The specification of the $\mathtt{corrupt}(\cdot)$ query for group signatures systems is as follows.

- $\mathtt{corrupt}(\cdot)$: on input $u \in \mathbb{N}$, the challenger outputs the secret key of the user identified by u, i.e. $gsk[u]$.

We now show why certain privacy notions do not apply to group signature systems, while others are equivalent.

Lemma 3. *No group signature system provides* $\mathsf{SA}^*, \mathsf{PS}^*, \mathsf{SA}^+$, *or* PS^+. *Moreover, for group signature systems,* SA° *and* PS° *are equivalent, and* SA *and* PS *are distinct, privacy notions.*

Proof. No group signature system provides $\mathsf{SA}^*, \mathsf{PS}^*, \mathsf{SA}^+, \mathsf{PS}^+$ because the signed message is published along with its group signature; $\mathcal{A}$ can trivially win an $\mathbf{Exp}^{\mathrm{priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment of providing different messages in the left and the right world. SA° and PS° are equivalent by Lemma 4, and SA and PS are distinct by Remark 3. □

Full anonymity Let us briefly revisit the definition of 'full anonymity' as defined in [5] and examine how it relates to the privacy notions that apply to online systems. Full anonymity is defined by means of an FA-experiment between an adversary $\mathcal{A}_{\mathsf{FA}}$ and a challenger, which proceeds as follows. Initially, the adversary is given gsk and gpk, and access to an opening oracle $\mathsf{Open}(gmsk, \cdot, \cdot)$ that, on input a message/signature pair (m, σ), outputs $\mathsf{Open}(gmsk, m, \sigma)$. At some point in time, the adversary outputs a triple (u_0, u_1, m') and the challenger returns $\sigma' = \mathsf{GSig}(gsk[u_b], m')$, where $b \in \{0, 1\}$ is chosen uniformly at random. The adversary is then required to output a guess for b; before doing this, it may again query the $\mathsf{Open}(gmsk, \cdot, \cdot)$ oracle, albeit not on σ'. The adversary wins if its guess is correct, and the system is said to provide 'full anonymity', denoted FA, if no adversary can win the game with non-negligible advantage over random guessing.

Lemma 4. FA*, computational* SA°*, and computational* PS°*, are equivalent.*

Proof. We first show that PS° implies FA. An $\mathcal{A}_{\mathsf{PS}^\circ}$ adversary with access to an $\mathcal{A}_{\mathsf{FA}}$ adversary starts by corrupting all users, obtaining their secret keys, which it feeds into $\mathcal{A}_{\mathsf{FA}}$. $\mathcal{A}_{\mathsf{FA}}$ makes only one query which $\mathcal{A}_{\mathsf{PS}^\circ}$ passes on to the challenger and gives the response back to $\mathcal{A}_{\mathsf{FA}}$. The restriction $P_{f_0} = P_{f_1}$ is satisfied, since any two functions with a singleton domain induce the same partition on their domain. $\mathcal{A}_{\mathsf{PS}^\circ}$ answers in the same way as $\mathcal{A}_{\mathsf{FA}}$.

We show that FA also implies SA° by constructing an adversary $\mathcal{A}_{\mathsf{FA}}$ that has a non-negligible advantage in the FA-experiment, given black-box access to an adversary $\mathcal{A}_{\mathsf{SA}^\circ}$ with non-negligible advantage in a (computational) SA°-experiment. $\mathcal{A}_{\mathsf{FA}}$ proceeds as follows. It uniformly at random selects a value $i \in \mathbb{N}$ such that $1 \leq q(k)$, where $q(k)$ is the upper bound on the number of queries that $\mathcal{A}_{\mathsf{SA}^\circ}$ may issue. Using its knowledge of gsk, it answers $\mathcal{A}_{\mathsf{SA}^\circ}$'s first $i-1$ $\mathtt{input}((u_0, m), (u_1, m))$ queries with $\mathsf{GSig}(gsk[u_0], m)$. Before answering $\mathcal{A}_{\mathsf{SA}^\circ}$'s ith $\mathtt{input}((u_0, m'), (u_1, m'))$ query, it queries the challenger with the triple (u_0, u_1, m') with values taken from $\mathcal{A}_{\mathsf{SA}^\circ}$'s query. $\mathcal{A}_{\mathsf{FA}}$ returns the challenger's answer σ' to $\mathcal{A}_{\mathsf{SA}^\circ}$. Using its knowledge of gsk, it answers $\mathcal{A}_{\mathsf{SA}^\circ}$'s remaining $\mathtt{input}((u_0, m), (u_1, m))$ queries with $\mathsf{GSig}(gsk[u_1], m)$, and finally outputs the same value as $\mathcal{A}_{\mathsf{SA}^\circ}$. Using a standard hybrid argument [22], it can be shown that $\mathcal{A}_{\mathsf{FA}}$'s success probability is $(1/2) + \delta/q$, where q and δ are the number of queries issued by $\mathcal{A}_{\mathsf{SA}^\circ}$ and $\mathcal{A}_{\mathsf{SA}^\circ}$'s advantage, respectively. Since, as shown in Section 3.3, SA° implies PS°, a group signature system that provides FA also provides PS°. □

The fact that there exists only a single (computational) forward/backward privacy notion for group signatures, explains, perhaps, why [5] claims that 'anonymity and unlinkability are technically the same property'.

Remark 3. From our framework it is now obvious that weaker privacy notions for group signatures exist; it is possible to refrain from forward/backward privacy, and optionally in addition tolerate the group signatures of the same signer being linkable. Traceable group signature schemes were to our knowledge first considered in [27]. We modify a traceable scheme from [7] to construct an instance of a group signature scheme that provides PS but not SA. The required modification is minor, as it merely consists in setting a particular parameter of the scheme to one. We now briefly review the modified scheme; for a complete description see [7]. The scheme uses a bilinear group pair (G_1, G_2) consisting of cyclic groups G_1 and G_2 of prime order p with an efficiently computable isomorphism from G_2 to G_1, and an efficiently computable non-degenerate bilinear map $e : G_1 \times G_2 \to G_t$. For (G_1, G_2) the strong Diffie-Hellman assumption (see. [7]) has to hold. A public group key gpk is given by a triple of group elements (g_1, g_2, g_2^γ), where $g_1 \in G_1$, $g_2 \in G_2$ are randomly chosen from the respective groups and act as generators, and γ is secretly and uniformly at random chosen from $\mathbb{Z}_p$. Then a private signing key for a user U_i is given by $(A_i = g_1^{1/(\gamma + x_i)}, x_i)$ for a uniformly at random chosen element $x_i \in \mathbb{Z}_p$. Furthermore, a hash function $H : \{0, 1\}^* \to G_1 \times G_2$ is given. The signing procedure is as follows:

1. $(u, v) \leftarrow H(gpk)$
2. Choose $\alpha \leftarrow \mathbb{Z}_p$ uniformly at random and compute $T_1 \leftarrow u^\alpha$, $T_2 \leftarrow A_i v^\alpha$

3. Compute $c, s_\alpha, s_x, s_\delta$ as a witness indistinguishable proof of knowledge for correct computation of T_1, T_2 with respect to the private key A_i. This is done with Fiat-Shamir heuristic [17] and involves the message being signed.

The signature of a message is then given by $(T_1, T_2, c, s_\alpha, s_x, s_\delta)$. Now any two group signatures of the same signer can be linked by computing $e(A_i, u) = e(T_2, u)/e(T_1, v)$ which is a value that, since the parameters u, v are common to the entire group, depends only on the signer's private key. Since this value cannot be traced back to any particular public key, some privacy remains.

Remark 4. We are not aware of any group signature scheme that provides SA but not FA at the same time, i.e. a scheme without forward-/backward privacy. The following example, however, demonstrates the existence of such a scheme. Consider a group signature scheme that provides FA. We modify this scheme as follows. Every signer is given a pseudorandom number generator whose seed is part of the user's secret key and, in order to sign a message, the user replaces all random choices by pseudo-randomness. As a result, every user behaves deterministically and, as long as the adversary does not know a user's seed, the produced signatures are computationally indistinguishable from those based on true randomness. Moreover, once the adversary calls `corrupt` on a user and learns his seed, all past and future signatures of this user become linkable; the modified scheme no longer provides forward-/backward privacy. In fact, it provides computational SA. Note that this scheme might well apply to smart-card group signature implementations where replacing randomness by pseudo-randomness is a common option.

4.2 Anonymous communication

Anonymous communication systems are modelled as protocols that transmit messages from senders to receivers. The input to an anonymous communication system is a sequence of triples of the form $(\sigma, \rho, m) \in \mathbb{N} \times \mathbb{N} \times \mathcal{M}$, where $\sigma, \rho \in \mathbb{N}$ are identifiers of the sender and the intended recipient, respectively, $\mathcal{M}$ is the system's message space, and $m \in \mathcal{M}$ is the message that is to be transmitted from σ to ρ. The output that is associated to an input triple of this form, is the bitstring that the system produces as a result of this input, and that the adversary can observe.

For anonymous communication systems we define two variants of the base experiment $\mathbf{Exp}^{\mathrm{priv-b}}_{\Phi^A,\mathcal{A}}(k)$, depending on whether the experiment is intended to capture the privacy of senders or the privacy of recipients. In particular, the variant that captures sender privacy is denoted by $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$, and the variant that captures recipient privacy by $\mathbf{Exp}^{\mathrm{R-priv-b}}_{\Phi^A,\mathcal{A}}(k)$. In both variants, the parameter space is $A_{\mathtt{ac}} = \mathcal{M} \times \mathbb{N}$. The difference between the two variants is the way in which the challenger assigns the sender and receiver roles to the users indicated in an $\mathtt{input}((\cdot,\cdot),(\cdot,\cdot))$ query; in all other respects the two variants are identical to the base experiment.

Definition 6. *On reception of an* $\mathtt{input}((u_0,(m_0,u_0')),(u_1,(m_1,u_1')))$ *query in the context of an* $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ *(resp.* $\mathbf{Exp}^{\mathrm{R-priv-b}}_{\Phi^A,\mathcal{A}}(k)$*) experiment, the challenger first increases the input counter* $\mathtt{c}$ *by one, and then remembers* $(u_b,(m_b,u_b'))$ *(resp* $(u_b',(m_b,u_b))$*) as* $(u_{\mathtt{c}},\alpha_{\mathtt{c}})$.

In other words, the parameter $\alpha = (m, u) \in A_{\mathtt{ac}}$ either specifies a message together with (the identifier of) its intended recipient (in the context of an $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment), or a message together with (the identifier of) its sender (in the context of an $\mathbf{Exp}^{\mathrm{R-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment). We now extend our generic definition for the context of anonymous communication.

Definition 7. *An anonymous communication system* $\Phi^{A_{\mathtt{ac}}}$ *is said to unconditionally (resp. statistically, computationally) provide 'sender-X', denoted* S/X *(resp. 'recipient-X', denoted* R/X*) for some privacy notion* $X \in \{Y^*, Y^\circ, Y^+, Y\}$ *where* $Y \in \{\mathsf{SA}, \mathsf{PH}, \mathsf{SU}, \mathsf{WU}, \mathsf{PS}, \mathsf{AN}, \mathsf{WA}\}$*, if and only if it unconditionally (resp. statistically, computationally) provides* X *with respect to an* $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ *(resp.* $\mathbf{Exp}^{\mathrm{R-priv-b}}_{\Phi^A,\mathcal{A}}(k)$*) experiment.*

It trivially follows from the definition that $\mathsf{S/SA}^+$ and $\mathsf{R/SA}^+$, as well as $\mathsf{S/SA}^*$ and $\mathsf{R/SA}^*$, are equivalent notions. We define one more privacy notion, namely unlinkability, denoted UL. UL is specific to anonymous communication systems, and, like SA^*, its sender and recipient versions are equivalent. Unlinkability is the notion that ensures that $\mathcal{A}$ cannot learn anything about f beyond what follows from knowledge of how many messages each sender sent, and how many messages each receiver received. Let $A_0 = ((\cdot, u'_{0,1}), (\cdot, u'_{0,2}), \ldots, (\cdot, u'_{0,x}))$ and $A_1 = ((\cdot, u'_{1,1}), (\cdot, u'_{1,2}), \ldots, (\cdot, u'_{1,x}))$ denote parameter sequences issued by the adversary during an $\mathbf{Exp}^{\mathrm{priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment.

Definition 8. *An anonymous communication system $\Phi^{A_{\mathrm{ac}}}$ is said to unconditionally (resp. statistically, computationally) provide privacy notion UL^* (resp.$\mathsf{UL}^\circ, \mathsf{UL}^+, \mathsf{UL}$}, called unlinkability, if and only if it unconditionally (resp. statistically, computationally) provides WU^* (resp. $\mathsf{WU}^\circ, \mathsf{WU}^+, \mathsf{WU}$) with respect to an $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ and an $\mathbf{Exp}^{\mathrm{R-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment where, for all $i \in \{1, 2, \ldots, x\}$, $u'_{0,i} = u'_{1,i}$.*

Existing notions We briefly revisit the privacy notions defined in [24] in order to examine how they relate to the ones defined above. [24] defines privacy by means of an experiment between an adversary and a challenger. The adversary specifies in advance two collections C^0 and C^1 of triples of the form $(\sigma, \rho, m) \in \mathbb{N}^2 \times \mathcal{M}$.[2] The two collections are then given to the challenger, which selects a bit $b \in \{0, 1\}$ uniformly at random, and simulates $\Phi^{A_{\mathrm{ac}}}$ on input the triples in C^b. The adversary, given $\Phi^{A_{\mathrm{ac}}}$'s output, then produces a guess g for b and wins if and only if $g = b$; its advantage is defined in the usual way.

Let $S^b = \{\sigma \in \mathbb{N} : (\sigma, \cdot, \cdot) \in C^b\}$ and $R^b = \{\rho \in \mathbb{N} : (\cdot, \rho, \cdot) \in C^b\}$ denote the set of senders and receivers according to C^b. For all $\sigma \in S^b$ (resp. $\rho \in R^b$), we denote by $\mathsf{sent}^b_\sigma = (\uplus m \in \mathcal{M} : (\sigma, \cdot, m) \in C^b)$ (resp. $\mathsf{rcvd}^b_\rho = (\uplus m \in \mathcal{M} : (\cdot, \rho, m) \in C^b)$) the multiset of messages sent by σ (resp. received by ρ) according to C^b. The different privacy notions defined in [24] arise due to restrictions imposed on the adversary in the construction of C^0 and C^1. In particular, an anonymous communication system is said to provide privacy notion $N \in \{\overline{\mathsf{SUL}}, \overline{\mathsf{RUL}}, \overline{\mathsf{UL}}, \overline{\mathsf{SA}}, \overline{\mathsf{RA}}, \overline{\mathsf{SA}^*}, \overline{\mathsf{RA}^*}, \overline{\mathsf{SRA}}, \overline{\mathsf{UO}}\}$ if no adversary, when restricted to choose C^0 and C^1 such that the conditions shown in Table 1 are satisfied, has a non-negligible advantage in the above experiment.

Comparison to existing notions The adversarial model in [24] does not consider corrupted users, and does not consider adaptive adversaries. Translated to our system model, this amounts to the setting where $\mathcal{A}$ issues only a single `nextBatch` query, and no `corrupt`$(\cdot)$ queries. Due to this discrepancy of the adversarial models, the privacy notions defined in this paper are not directly comparable to the ones defined in [24]. If, however, $\mathcal{A}$ is restricted to observe only a single batch and is allowed no corruptions, then following notions are equivalent.

Lemma 5. *If, during an $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ or $\mathbf{Exp}^{\mathrm{R-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment, $\mathcal{A}$ does not issue any* `corrupt`$(\cdot)$ *queries and at most a single* `nextBatch` *query, then $\overline{\mathsf{SUL}}$ and $\mathsf{S/WU}$, $\overline{\mathsf{RUL}}$ and $\mathsf{R/WU}$, $\overline{\mathsf{UL}}$ and UL, $\overline{\mathsf{SA}}$ and $\mathsf{S/SA}$, $\overline{\mathsf{RA}}$ and $\mathsf{R/SA}$, $\overline{\mathsf{SA}^*}$ and $\mathsf{S/WU}^+$, $\overline{\mathsf{RA}^*}$ and $\mathsf{R/WU}^+$, and $\overline{\mathsf{RA}^*}$ and $(\mathsf{S/R})\mathsf{SA}^+$, are equivalent privacy notions.*

Proof. Consider an adversary $\mathcal{A}_{\overline{\mathsf{SUL}}}$. We construct an adversary $\mathcal{A}_{\mathsf{WU}}$ that wins an $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment if and only if $\mathcal{A}_{\overline{\mathsf{SUL}}}$ wins. Let C^0 and C^1 denote the collections output by $\mathcal{A}_{\overline{\mathsf{SUL}}}$. Due to the applicable restrictions $S = S^0 = S^1$, $R = R^0 = R^1$, $|\mathsf{sent}^0_\sigma| = |\mathsf{sent}^1_\sigma|$ for all $\sigma \in S$, and $\mathsf{rcvd}^0_\rho = \mathsf{rcvd}^1_\rho$ for all $\rho \in R$ (see Table 1), for each triple $(\sigma_0, \rho_0, m_0) \in C^0$ there exists exactly one 'corresponding' triple in C^1, i.e. a triple (σ_1, ρ_1, m_1) such that $\rho_1 = \rho_0$ and $m_1 = m_0$. For each triple in $(\sigma_0, \rho_0, m_0) \in C_0$, $\mathcal{A}_{\mathsf{WU}}$ issues the query `input`$((\sigma_0, (\rho_0, m_0)), (\sigma_1, (\rho_1, m_1)))$ where σ_1, ρ_1

[2] In [24] these collections are called 'message matrices', and are encoded as matrices.

Privacy notion	Label	Conditions
Sender Unlinkability	$\overline{\mathsf{SUL}}$	$S = S^0 = S^1$, $R = R^0 = R^1$, $\forall \sigma \in S,\ \lvert\mathsf{sent}^0_\sigma\rvert = \lvert\mathsf{sent}^1_\sigma\rvert$, and $\forall \rho \in R,\ \mathsf{rcvd}^0_\rho = \mathsf{rcvd}^1_\rho$
Receiver Unlinkability	$\overline{\mathsf{RUL}}$	$S = S^0 = S^1$, $R = R^0 = R^1$, $\forall \sigma \in S,\ \mathsf{sent}^0_\sigma = \mathsf{sent}^1_\sigma$, $\forall \rho \in R,\ \lvert\mathsf{rcvd}^0_\rho\rvert = \lvert\mathsf{rcvd}^1_\rho\rvert$, and
Unlinkability	$\overline{\mathsf{UL}}$	$S = S^0 = S^1$, $R = R^0 = R^1$, $\forall \sigma \in S,\ \lvert\mathsf{sent}^0_\sigma\rvert = \lvert\mathsf{sent}^1_\sigma\rvert$, and $\forall \rho \in R,\ \lvert\mathsf{rcvd}^0_\rho\rvert = \lvert\mathsf{rcvd}^1_\rho\rvert$
Sender Anonymity	$\overline{\mathsf{SA}}$	$R = R^0 = R^1$ and, $\forall \rho \in R\ \mathsf{rcvd}^0_\rho = \mathsf{rcvd}^1_\rho$
Receiver Anonymity	$\overline{\mathsf{RA}}$	$S = S^0 = S^1$ and, $\forall \sigma \in S,\ \mathsf{sent}^0_\sigma = \mathsf{sent}^1_\sigma$
Strong Sender Anonymity	$\overline{\mathsf{SA}^*}$	$R = R^0 = R^1$ and, $\forall \rho \in R\ \lvert\mathsf{rcvd}^0_\rho\rvert = \lvert\mathsf{rcvd}^1_\rho\rvert$
Strong Receiver Anonymity	$\overline{\mathsf{RA}^*}$	$S = S^0 = S^1$ and, $\forall \sigma \in S,\ \lvert\mathsf{sent}^0_\rho\rvert = \lvert\mathsf{sent}^1_\rho\rvert$
Sender-Receiver Anonymity	$\overline{\mathsf{RA}^*}$	$\lvert C^0\rvert = \lvert C^1\rvert$
Unobservability	$\overline{\mathsf{UO}}$	none

Table 1. Conditions according to privacy definitions in [24].

and m_1 are the values from the corresponding triple in C^1. $\mathcal{A}_{\mathsf{WU}}$ then issues a `nextBatch` query, forwards the challenger's output to $\mathcal{A}_{\overline{\mathsf{SUL}}}$, and, finally outputs a guess that is identical to $\mathcal{A}_{\overline{\mathsf{SUL}}}$'s guess. Clearly, $\mathcal{A}_{\mathsf{WU}}$ wins if and only if $\mathcal{A}_{\overline{\mathsf{SUL}}}$ wins.

Consider the adversary $\mathcal{A}_{\mathsf{WA}}$ of an $\mathbf{Exp}^{\mathrm{S-priv-b}}_{\Phi^A,\mathcal{A}}(k)$ experiment. We construct an adversary $\mathcal{A}_{\overline{\mathsf{SUL}}}$ that wins if and only if $\mathcal{A}_{\mathsf{WA}}$ wins. For every $\mathtt{input}((\sigma_0,(\rho_0,m_0)),(\sigma_1,(\rho_1,m_1)))$ query issued by $\mathcal{A}_{\mathsf{WA}}$, $\mathcal{A}_{\overline{\mathsf{SUL}}}$ adds the triple $(\sigma_0,(\rho_0,m_0))$ to C^0 and the triple $(\sigma_1,(\rho_1,m_1))$ to C^1. When $\mathcal{A}_{\mathsf{WA}}$ issues the `nextBatch` query, $\mathcal{A}_{\overline{\mathsf{SUL}}}$ starts its experiment with C^0 and C^1. Note that, due to the restrictions that apply in the experiment of $\mathcal{A}_{\mathsf{WA}}$, the collections C^0 and C^1, too, satisfy the required restrictions. $\mathcal{A}_{\overline{\mathsf{SUL}}}$ then forwards the challenger's output to $\mathcal{A}_{\mathsf{WA}}$, and, finally outputs a guess that is identical to $\mathcal{A}_{\mathsf{WA}}$'s guess. Clearly, $\mathcal{A}_{\overline{\mathsf{SUL}}}$ wins if and only if $\mathcal{A}_{\mathsf{WA}}$ wins. Thus, $\overline{\mathsf{SUL}}$ and $\mathsf{S/WU}$ are equivalent privacy notions. Showing the validity of the other equivalences is analogous. □

Remark 5. The above privacy notions form a hierarchy, described in [24], that is separate from the one described in Section 3. Moreover, [24] demonstrates that one can construct anonymous communication systems that offer a particular privacy notion by appropriately augmenting a system that provides a weaker notion, with encryption techniques and/or dummy traffic. Since, according to the model in [24], the adversary may observe only a single communication batch, these transformations do not necessarily suffice in the face an adversary that may adaptively influence the system over multiple communication batches, i.e. in the model considered in this paper.

Since in our model, $\mathcal{A}$ may issue multiple `nextBatch` queries, the notions $\mathsf{S/WU}$, $\mathsf{R/WU}$, UL, $\mathsf{S/SA}$, $\mathsf{R/SA}$, $\mathsf{S/WU}^+$, $\mathsf{R/WU}^+$, and $\mathsf{(S/R)SA}^+$, are all strictly stronger than $\overline{\mathsf{SUL}}$, $\overline{\mathsf{RUL}}$, $\overline{\mathsf{UL}}$, $\overline{\mathsf{SA}}$, $\overline{\mathsf{RA}}$,$\overline{\mathsf{SA}^*}$,$\overline{\mathsf{RA}^*}$, and $\overline{\mathsf{RA}^*}$, respectively. Consider, for example, an anonymous communication system that provides notion $\overline{\mathsf{RA}}$, i.e. a system where, for an adversarially chosen batch of communications (where certain conditions hold), the adversary may be able to determine which messages were received by which receivers, but no information beyond this. In contrast to this, the system would only provide notion $\mathsf{S/SA}$ if it does not leak any such information even for multiple, adversarially and adaptively chosen batches of communication (where certain conditions hold). This suggests that an anonymous communication system provides a privacy notion in $\{\mathsf{S/WU}, \mathsf{R/WU}, \mathsf{UL}, \mathsf{S/SA}, \mathsf{R/SA}, \mathsf{S/WU}^+, \mathsf{R/WU}^+, \mathsf{(S/R)SA}^+\}$ only if it is, effectively immune to 'disclosure' (also known as 'hitting set') attacks [2, 26], while privacy notions in $\{\overline{\mathsf{SUL}}, \overline{\mathsf{RUL}}, \overline{\mathsf{UL}}, \overline{\mathsf{SA}}, \overline{\mathsf{RA}}, \overline{\mathsf{SA}^*}, \overline{\mathsf{RA}^*}, \overline{\mathsf{SRA}}\}$ can be achieved without such immunity.

4.3 Voting systems

A typical voting process has, among other things, a phase where users cast their votes, called the 'voting phase', and a phase where the results of the elections are computed and published, called the 'tallying phase'. Although secret voting systems have to fulfill a multitude of requirements, in this paper we focus only on *ballot secrecy*, since this is a privacy notion. Moreover, we assume that each eligible voter is allowed to cast at most one vote.[3]

Voting systems are modelled as follows. During the voting phase, the system accepts pairs $(u, c) \in \mathbb{N} \times \mathcal{C}$, where $\mathbb{N}$ is the set of (the identifiers of) the eligible voters, and $\mathcal{C}$ is the set of candidates in the election. The adversary can cause a ballot to be cast by issuing an `input` query, and indicate the end of the voting phase by issuing a `nextBatch` query. The bitstring β that the system produces as part of its output contains the tally and additional information that may be needed for verification. Since no further voting is possible after the tally is computed, voting systems produce only a single batch.

Ballot secrecy can be modelled in two ways within our framework. The first, perhaps more intuitive one, is as follows. First, the parameter space of voting systems is defined as $A^1_{\mathtt{vs}} = \mathcal{C}$. The challenger, on reception of an $\mathtt{input}((u_0, c_0), (u_1, c_1))$ query, then invokes the system on input (u_b, c_b). On reception of a $\mathtt{corrupt}(u)$ query, the challenger returns the secret information held by u including the candidate u voted for (if any). Some voting schemes, particularly those that are based on blind signatures (e.g. [20]) fit this model.

Lemma 6. *Assuming that each eligible voter may vote at most once, the strongest achievable privacy notion for voting schemes is, in the above modelling, unconditional* PH.

Proof. The tally reveals the number of participants, as this matches the total number of ballots. Thus, SA-distinguishable functions can be distinguished from the system output β. □

Unfortunately, not all voting systems can be modelled in the above way. This is because the elements produced by certain systems are homomorphically encrypted ballots whose correspondence to voters is not at all hidden; we call the systems that produce such elements 'homomorphic voting systems' [1].[4] Homomorphic voting systems provide privacy by ensuring that the identity of the candidate that is encoded in each (encrypted) ballot remains hidden.

Remark 6. A basic homomorphic voting scheme operates as follows. Each element it produces is an encryption of the identity of the candidate that was selected in the corresponding invocation, where encryption is performed using an asymmetric homomorphic encryption scheme. In the tallying phase, it first computes the election result based on the homomorphism of the encrypted elements, and then outputs all elements (i.e. encrypted ballots), together with the election result and a proof of correct decryption.

Homomorphic voting systems are modelled in our framework through a reversal of the roles of voters and candidates. That is, instead of treating 'ballot secrecy' as a privacy property concerning voters, it is treated as a privacy property concerning candidates. A system is then said to provide ballot secrecy if it hides the identities of the users that have voted for any given candidate; it is easy to see that this is equivalent to the case where it hides the identity of the candidate any given user has voted for. This reversal, however, enables us to alternatively define $\mathcal{A}$'s interaction with the challenger based on candidates rather than voters. In particular, the parameter space for voting system as $A^2_{\mathtt{vs}} = \mathbb{N}$, where $\mathbb{N}$ is the identifier set of eligible voters, and $\mathcal{C}$ is used as the set of users. In particular, on reception of an $\mathtt{input}((c_0, u_0), (c_1, u_1))$ query, the challenger invokes the system

[3] We are unaware of any formal treatments of ballot secrecy in the literature; our formal model below follows the intuition provided by the notion of 'perfect ballot secrecy' that is informally described in [28].

[4] These systems do not hide this correspondence in order to provide 'individual verifiability', a property that enables each voter to verify that his own vote has been accounted for in the tally.

$\Phi^{A^2_{\text{vs}}}$ on input (u_b, c_b) and, on reception of a `corrupt(`c`)` query, the challenger returns the set of users that have voted for c so far (if any). In order to break privacy, $\mathcal{A}$ would have to identify two (encrypted) ballots for the same candidate, or reveal the candidate of one ballot. We now show why the strongest possible notion for ballot secrecy is weak unlinkability.

Lemma 7. *In the above modelling, computational* WU *is the strongest achievable ballot secrecy notion for homomorphic voting systems.*

Proof. Homomorphic encryption schemes hide the plaintext only from computationally bounded adversaries, and the tally reveals Q_{f_b}. □

If a homomorphic voting system is computationally stateless as defined in Section 3.2, then it provides computational WA. That is an immediate consequence of Lemma 1 and Lemma 7. Voting schemes that are based on homomorphic encryption are computationally stateless as long as the underlying encryption scheme is IND-CPA [22] and the order of elements is uniformly permuted.

5 Conclusions and open questions

We presented an application-agnostic hierarchy of privacy notions that describe potentially different degrees to which the correspondence between digital elements and the users that cause their appearance remains hidden from an adversary. Previously isolated privacy notions pertaining to group signatures, anonymous communication, and voting systems have been placed into this hierarchy, and thereby effectively made comparable. It is possible that privacy definitions pertaining to other system types, such as anonymous credentials, data anonymisation systems, and sensor information systems, can also be placed into our framework. Examining this possibility is subject of future research.

Our framework provides valuable insights into the relations and structure of different privacy notions, and highlights a largely unexplored space of such notions. Exemplarily, we identified two new notions for group signatures and pointed out how group signatures that match these definitions look like. Identifying useful schemes providing other 'new' notions, perhaps by trading off privacy against other features, is subject of future research. Of particular interest are techniques that transform systems achieving a given privacy notion into systems that provide another, perhaps stronger one in the adaptive adversarial model considered in this paper. We expect that the framework will also be useful in the construction and analysis of 'multi-layer' privacy protecting systems, i.e. systems that combine, for example, anonymous communication with group signing.

Finally, constructing 'soft', probabilistic privacy metrics for each of the notions in our framework is subject of current research. Such metrics will enable us to compare privacy protecting systems with considerably higher granularity than is possible with definitions that are based on asymptotic polynomial indistinguishability.

6 Acknowledgments

The authors would like to thank the anonymous reviewers for the insightful comments. This paper describes work undertaken partly in the context of the 'Integrating the Physical with the Digital World of the Network of the Future' (SENSEI) project (www.sensei-project.eu), and partly in the 'Secure Widespread Identities for Federated Telecommunications' (SWIFT) project (www.ist-swift.org). SENSEI and SWIFT are collaborative projects supported by the 7th European Framework Programme, with contract numbers 215923 and 215832, respectively.

References

1. B. Adida. Advances in cryptographic voting systems. PhD thesis, Massachusetts Institute of Technology, 2006.
2. D. Agrawal and D. Kesdogan. Measuring anonymity: The disclosure attack. *IEEE Security & Privacy*, 1(6):27–34, 2003.
3. C. Andersson and R. Lundin. On the fundamentals of anonymity metrics. In *The Future of Identity in the Information Society*, IFIP International Federation for Information Processing. Springer Science & Business Media, 2008.
4. M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science (FOCS'97)*, pages 394–403, 1997.
5. M. Bellare, D. Micciancio, and B. Warinschi. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In E. Biham, editor, *Advances in Cryptology – EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4–8, 2003, Proceedings*, volume 2656 of *Lecture Notes in Computer Science*, pages 614–629. Springer, 2003.
6. R. Berman, A. Fiat, and A. Ta-Shma. Provable unlinkability against traffic analysis. In A. Juels, editor, *Financial Cryptography, 8th International Conference, FC 2004, Key West, FL, USA, February 9–12, 2004. Revised Papers*, volume 3110 of *Lecture Notes in Computer Science*, pages 266–280. Springer Verlag, Berlin, February 2004.
7. D. Boneh and H. Shacham. Group signatures with verifier-local revocation. In *ACM Conference on Computer and Communications Security, CCS 2004*, pages 168–177. ACM, 2004.
8. J. Camenisch and A. Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In B. Pfitzmann, editor, *Advances in Cryptology — EUROCRYPT 2001, International Conference on the Theory and Application of Cryptographic Techniques, Innsbruck, Austria, May 6–10, 2001, Proceedings*, volume 2045 of *Lecture Notes in Computer Science*, pages 93–118. Springer Verlag, Berlin, 2001.
9. D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.
10. S. Clauß. A framework for quantification of linkability within a privacy-enhancing identity management system. In G. Müller, editor, *Emerging Trends in Information and Communication Security, International Conference, ETRICS 2006, Freiburg, Germany, June 6-9, 2006, Proceedings*, volume 3995 of *Lecture Notes in Computer Science*, pages 191–205. Springer Verlag, 2006.
11. S. Clauß and S. Schiffner. Structuring anonymity metrics. In *DIM '06: Proceedings of the second ACM workshop on Digital identity management*, pages 55–62, New York, NY, USA, 2006. ACM Press.
12. C. Díaz. Anonymity metrics revisited. In *Anonymous Communication and its Applications, number 05411 in Dagstuhl Seminar Proceedings*, 2005.
13. C. Díaz, J. Claessens, S. Seys, and B. Preneel. Information theory and anonymity. In B. Macq and J. Quisquater, editors, *Proceedings of the 23rd Symposium on Information Theory in the Benelux*, pages 179–186, 2002.
14. C. Díaz, S. Seys, J. Claessens, and B. Preneel. Towards measuring anonymity. In R. Dingledine and P. F. Syverson, editors, *Privacy Enhancing Technologies, Second International Workshop, PET 2002, San Francisco, CA, USA, April 14–15, 2002, Revised Papers*, number 2482 in Lecture Notes in Computer Science, pages 54–68. Springer Verlag, Berlin, 2002.
15. M. Edman, F. Sivrikaya, and B. Yener. A combinatorial approach to measuring anonymity. In *Proceedings of the 2007 IEEE International Conference on Intelligence and Security Informatics*. IEEE, 2007.
16. J. Feigenbaum, A. Johnson, and P. Syverson. Probabilistic analysis of onion routing in a black-box model. In *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, pages 1–10. ACM Press, 2007.
17. A. Fiat and A. Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In A. M. Odlyzko, editor, *Advances in cryptology – CRYPTO '86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer, 1988.
18. L. Fischer, S. Katzenbeisser, and C. Eckert. Measuring unlinkability revisited. In M. Winslett and R. H. to be published, editors, *Proceedings of the 2008 ACM Workshop on Privacy in the Electronic Society, WPES 2008, Alexandria, VA, USA, October 27, 2008*, pages 00–00. ACM, 2008.

19. M. Franz, B. Meyer, and A. Pashalidis. Attacking unlinkability: The importance of context. In N. Borisov and P. Golle, editors, *Privacy Enhancing Technologies, 7th International Symposium, PET 2007 Ottawa, Canada, June 20-22, 2007, Revised Selected Papers*, volume 4776 of *Lecture Notes in Computer Science*, pages 1–16. Springer Verlag, Berlin, 2007.
20. A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. In *ASIACRYPT '92*, number 718 in Lecture Notes in Computer Science, pages 244–251. Springer, 1993.
21. B. Gierlichs, C. Troncoso, C. Díaz, B. Preneel, and I. Verbauwhede. Revisiting a combinatorial approach toward measuring anonymity. In *Proceedings of the 2008 ACM Workshop on Privacy in the Electronic Society, WPES 2008, Alexandria, VA, USA, October 27, 2008*, pages 111–116. ACM, 2008.
22. S. Goldwasser and S. Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.
23. J. Y. Halpern and K. R. O'Neill. Anonymity and information hiding in multiagent systems. *Journal of Computer Security*, 13(3):483–514, 2005.
24. A. Hevia and D. Micciancio. An indistinguishability-based characterization of anonymous channels. In N. Borisov and I. Goldberg, editors, *Privacy Enhancing Technologies Symposium, Eighth International Workshop, PET 2008, Leuven, Belgium, July 23–25, 2008, Revised Selected Papers*, volume 5134 of *Lecture Notes in Computer Science*, pages 24–43. Springer Verlag, Berlin, 2008.
25. D. Hughes and V. Shmatikov. Information hiding, anonymity and privacy: a modular approach. *Journal of Computer Security*, 12(1):3–36, 2004.
26. D. Kesdogan, D. Agrawal, and S. Penz. Limits of anonymity in open environments. In F. Petitcolas, editor, *Information Hiding, 5th International Workshop, IH 2002, Noordwijkerhout, The Netherlands, October 7–9, 2002, Revised Papers*, volume 2578 of *Lecture Notes in Computer Science*, pages 53–69. Springer Verlag, Berlin, 2003.
27. A. Kiayias, Y. Tsiounis, and M. Yung. Traceable signatures. In *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 571–589. Springer, 2004.
28. A. Kiayias and M. Yung. Self-tallying elections and perfect ballot secrecy. In D. Naccache and P. Paillier, editors, *5th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2002 Paris, France, February 12–14, 2002 Proceedings*, number 2274 in Lecture Notes in Computer Science, pages 141–158. Springer Verlag, Berlin, 2002.
29. G. Maitland, J. Reid, E. Foo, C. Boyd, and E. Dawson. Linkability in practical electronic cash design. In J. Pieprzyk, E. Okamoto, and J. Seberry, editors, *Information Security, Third International Workshop, ISW 2000, Wollongong, NSW, Australia, December 20–21, 2000, Proceedings*, volume 1975 of *Lecture Notes in Computer Science*, pages 149–163. Springer Verlag, 2000.
30. R. E. Newman, I. S. Moskowitz, P. Syverson, and A. Serjantov. Metrics for traffic analysis prevention. In R. Dingledine, editor, *Privacy Enhancing Technologies, Third International Workshop, PET 2003, Dresden, Germany, March 26–28, 2003, Revised Papers*, volume 2760 of *Lecture Notes in Computer Science*, pages 48–65. Springer Verlag, Berlin, 2003.
31. A. Pashalidis. Measuring the effectiveness and the fairness of relation hiding systems. In *Proceedings of the First International Workshop on Multimedia, Information Privacy and Intelligent Computing Systems*. IEEE Press, 2008.
32. A. Pfitzmann and M. Köhntopp. Anonymity, unobservability, and pseudonymity: A proposal for terminology. In H. Federrath, editor, *Designing Privacy Enhancing Technologies, International Workshop on Design Issues in Anonymity and Unobservability, Berkeley, CA, USA, July 25–26, 2000. Proceedings*, volume 2009 of *Lecture Notes in Computer Science*, pages 1–9. Springer, 2000.
33. A. Serjantov. *On the Anonymity of Anonymity Systems*. Phd thesis, University of Cambridge, 2004.
34. A. Serjantov and G. Danezis. Towards an information theoretic metric for anonymity. In R. Dingledine and P. F. Syverson, editors, *Privacy Enhancing Technologies, Second International Workshop, PET 2002, San Francisco, CA, USA, April 14–15, 2002, Revised Papers*, volume 2482 of *Lecture Notes in Computer Science*, pages 41–53. Springer Verlag, Berlin, 2002.
35. V. Shmatikov and M.-H. Wang. Measuring relationship anonymity in mix networks. In *WPES '06: Proceedings of the 5th ACM workshop on Privacy in electronic society*, pages 59–62, New York, NY, USA, 2006. ACM Press.
36. S. Steinbrecher and S. Köpsell. Modelling unlinkability. In R. Dingledine, editor, *Privacy Enhancing Technologies, Third International Workshop, PET 2003, Dresden, Germany, March 26-28, 2003, Revised Papers*, volume 2760 of *Lecture Notes in Computer Science*, pages 32–47. Springer Verlag, Berlin, 2003.
37. G. Tóth, Z. Hornák, and F. Vajda. Measuring anonymity revisited. In S. Liimatainen and T. Virtanen, editors, *Proceedings of the Ninth Nordic Workshop on Secure IT Systems*, pages 85–90, Espoo, Finland, November 2004.