Anonymous Authenticated Communication[⋆]

Fabio Banfi and Ueli Maurer

Department of Computer Science
ETH Zurich
8092 Zurich, Switzerland
{fabio.banfi,maurer}@inf.ethz.ch

Abstract. Anonymity and authenticity are apparently conflicting goals. Anonymity means *hiding* a party's identity whereas authenticity means *proving* a party's identity. So how can a set of senders authenticate their messages without revealing their identity? Despite the paradoxical nature of this problem, there exist many cryptographic schemes designed to achieve both goals simultaneously, in some form.

This paper provides a composable treatment of communication channels that achieve different forms of anonymity and authenticity. More specifically, three channel functionalities for many senders and one receiver are introduced which provide some trade-off between authenticity and anonymity (of the senders). For each of them, composably realizing it is proved to corresponds to the use of a certain type of cryptographic scheme, namely (1) a new type of scheme which we call *bilateral signatures* (syntactically related to designated verifier signatures), (2) *partial signatures*, and (3) *ring signatures*. This treatment hence provides composable semantics for (game-based) security definitions for these types of schemes. The results of this paper can be interpreted as the dual of the work by Kohlweiss et al. (PETS 2013), where composable notions for anonymous confidential communication were introduced and related to the security definitions of certain types of public-key encryption schemes, and where the treatment of anonymous authenticated communication was stated as an open problem.

Keywords: anonymous authenticity · composable security · bilateral signatures · partial signatures · anonymous signatures · ring signatures

[⋆] This is the full version of an article appearing in the proceedings of SCN 2022 which is available at https://doi.org/10.1007/978-3-031-14791-3_13.

Table of Contents

1 Introduction

1.1 Background and Motivation

When studying the security of public-key encryption (PKE) it is natural to consider a setting with one sender and many receivers, each generating its own key-pair and authentically transmitting the public key to the sender. Then a reasonable concern is whether ciphertexts subsequently generated by the sender for distinct receivers are (computationally) indistinguishable. This captures the intuitive notion of receiver anonymity from the standpoint of an eavesdropper, and is formalized by the security definition of *key-indistinguishability*, first proposed by Bellare et al. [BBDP01]. Almost a decade later, Abdalla et al. [ABN10] introduced another related notion for PKE, *robustness*, which intuitively captures the fact that ciphertexts can only be meaningfully decrypted using the correct corresponding private key, meaning that trying to decrypt with a wrong key results in an error.

It turns out that this further property is crucially needed in conjunction with key-indistinguishability in order to provide a "usable" form of anonymous PKE, and this has been highlighted by Kohlweiss et al. [KMO+13] by showing that both properties, together with IND-CCA security, are needed in order for a PKE scheme to enhance an anonymous insecure broadcast channel into an anonymous confidential broadcast channel. Importantly, their work also highlights how key-indistinguishability is a security notion that exclusively *preserves* anonymity, rather than "creating" it, whereas IND-CCA *lifts* insecurity to confidentiality, thus "creating" more security along the secrecy axis.

On the other hand, for the security of digital signature schemes (DSS) the natural setting to consider is the dual of the above: Many senders, each authentically publishing their public verification key, send messages to the same party, the receiver. Here too it is reasonable to consider anonymity (preservation), of the sender in this case, from the standpoint of an eavesdropper. But in this setting it is additionally also meaningful to study the stronger notion of anonymity from the standpoint of the receiver, that is, we might want the senders to remain anonymous not only towards an external attacker (the eavesdropper), but towards the receiver as well. We distinguish those two separate notions of anonymity in this setting as *external* and *internal*, respectively, where clearly the latter implies the former (but not vice versa). However, unlike for PKE, the situation is arguably more intricate for DSS; in fact, providing external anonymity alone already appears paradoxical: How can we guarantee (computational) indistinguishability of signatures, when in the usual application of DSS it is assumed that an eavesdropper has access to the corresponding message as well as all possible verification keys, and could therefore easily distinguish signatures generated with different keys by simply verifying the signature on the message against all keys?

A direct consequence of this apparent dilemma is that for the setting discussed above, the standard syntactic definition of a DSS cannot possibly achieve any meaningful form of anonymity, as we prove later within our framework. This is in fact the reason why in the cryptographic literature there exist a multitude of different security notions capturing various forms of anonymity in relation to syntactic modifications of the usual DSS definition. A non-exhaustive list of examples includes: group signatures [CvH91], ring signatures [RST01], anonymous signatures [YWDW06,Fis07,ZI09], and partial signatures [BD09,SY09].

In this work we take an alternative approach in order to treat the apparently oxymoronic problem of achieving anonymous authenticity: Instead of creating new syntactic modifications of DSS and ad-hoc game-based security definitions thereof, we begin from a more abstract point of view and identify possible applications where those two goals simultaneously come into play, and directly define security in a composable fashion, using the framework of constructive cryptography of Maurer and Renner [MR11,Mau12], requiring that a protocol realizes such an application relying on the public-key infrastructure (PKI). More precisely, we introduce three novel composable security notions for generic protocols, and then present concrete protocols satisfying each of those. The first protocol makes use of a novel cryptographic scheme, dubbed *bilateral signatures*, while the other two employ *partial signatures* and *ring signatures*, respectively.

1.2 Related Work

The goal of this work is to fill a blank in the composable treatment of anonymous *communication*.[1] In order to illustrate this, we need to first briefly and informally introduce some key concept that we will elaborate later.

As opposed to game-based security definitions, composable security definitions in constructive cryptography are simulation-based; on an abstract level, they are statements asserting that a cryptographic protocol constructs an ideal resource from a set of real ones, where a resource is a mathematical object capturing a certain functionality, and thus has interfaces through which parties, honest and dishonest, can interact. In more detail, for the simple setting with two honest parties—the sender and the receiver—and a dishonest party—the adversary—we consider a real resource $\mathbf{R}$ and an ideal resource $\mathbf{S}$, both having the same set of interfaces, S for the sender, R for the receiver, and E for the adversary. Then we say that a protocol π executed by the honest parties constructs $\mathbf{S}$ from $\mathbf{R}$, informally denoted as $\mathbf{R} \overset{\pi}{\Longmapsto} \mathbf{S}$, if there exists a simulator sim such that $\pi^{S,R}\,\mathbf{R}$ (the resource resulting from applying the protocol at the honest interfaces of the real resource) is indistinguishable from $\mathsf{sim}^{E}\,\mathbf{S}$ (the resource resulting from applying the simulator at the dishonest interface of the ideal resource).

Typical resources used in this simple setting are the *insecure channel* INS (which leaks everything the sender inputs to the adversary, and allows the latter to inject messages), the *authentic channel* AUT, the *confidential channel* CNF, and the secure (i.e., authentic and confidential) channel SEC, all allowing to send multiple messages. But in order to capture anonymity, we are interested in a setting where there are multiple parties. More concretely, we consider resources with n senders $S_1, \ldots, S_n$ and one receiver R (for which we use the intuitive notation ${}_{n\to 1}$), and resources with one sender and n receivers (for which we

[1] In particular, we are not directly considering (anonymous) *entity* authentication.

use the intuitive notation $_{1\to n}$). If one considers the above channels, a natural approach to extend them to this setting would be to simply compose them in parallel, but this would imply that the leakage now includes the identities of the sender S_i or the receiver R_i, since the individual channels are distinguishable by definition by the adversary. In the following table we summarize the guarantees provided by resources combining such channels (which we also denote as channels) in terms of what is leaked to the adversary relative to a message m input by a sender and whether the adversary can inject messages (such that the receiver can not distinguish whether the message was sent by the sender S or the adversary E).

Channel Name	**Symbol**	**Leaked**	**Inject**	**Symbol**	**Leaked**	**Inject**
Insecure	$\mathsf{INS}_{n\to 1}$	S_i, m	yes	$\mathsf{INS}_{1\to n}$	R_i, m	yes
Authentic	$\mathsf{AUT}_{n\to 1}$	S_i, m	no	$\mathsf{AUT}_{1\to n}$	R_i, m	no
Confidential	$\mathsf{CNF}_{n\to 1}$	$S_i, \|m\|$	yes	$\mathsf{CNF}_{1\to n}$	$R_i, \|m\|$	yes
Secure	$\mathsf{SEC}_{n\to 1}$	$S_i, \|m\|$	no	$\mathsf{SEC}_{1\to n}$	$R_i, \|m\|$	no

It seems natural that truly *anonymous* versions of these channels, that is, channels capturing sender and receiver anonymity, must *not* leak such identities to the adversary. Therefore we enhance the above channels with these guarantees (adding the prefix A- for *anonymous*), and summarize the new channels in the following table (note that in $\mathsf{A\text{-}AUT}_{n\to 1}$, $\mathsf{A\text{-}CNF}_{n\to 1}$, and $\mathsf{A\text{-}SEC}_{n\to 1}$, the receiver also obtains the identity S_i of the sender, along with the message m).

Channel Name	**Symbol**		**Leaked**	**Inject**
	Sender anon.	Receiver anon.		
Anonymous & Insecure	$\mathsf{A\text{-}INS}_{n\to 1}$	$\mathsf{A\text{-}INS}_{1\to n}$	m	yes
Anonymous & Authentic	$\mathsf{A\text{-}AUT}_{n\to 1}$	$\mathsf{A\text{-}AUT}_{1\to n}$	m	no
Anonymous & Confidential	$\mathsf{A\text{-}CNF}_{n\to 1}$	$\mathsf{A\text{-}CNF}_{1\to n}$	$\|m\|$	yes
Anonymous & Secure	$\mathsf{A\text{-}SEC}_{n\to 1}$	$\mathsf{A\text{-}SEC}_{1\to n}$	$\|m\|$	no

Other (non-anonymous) resources that we need in this setting are: $\mathsf{KEY}_{n\leftrightarrow 1}$, which provides each sender with a (different) *shared secret-key* with the receiver; $\mathsf{KEY}_{1\leftrightarrow n}$, which provides each receiver with a shared secret-key with the sender (in both resources, the adversary's interface is inactive); $\mathsf{1\text{-}AUT}_{n\to 1}$, which provides each sender with a (different) *single-use authentic channel* to the receiver; $\mathsf{1\text{-}AUT}_{1\leftarrow n}$, which provides the receiver with n (different) *single-use authentic channels*, one to each of the senders.

We stress again that we are considering anonymity *preservation*, therefore in the following we summarize the previous results from the literature in terms of constructions among the anonymous channels mentioned above (plus shared secret keys and one-time authentic channels). This means that both real and ideal core resources are anonymous, and hence the enhancement of security provided by a construction happens along a different axis (namely confidentiality, authenticity, or both).

- In the symmetric-key setting, two works provide sender anonymous constructions:
  - In [AHM+15], Alwen et al. show that for a simple protocol π_{pMAC} based on key-indistinguishable and unforgeable *probabilistic MAC* schemes,

$$[\mathsf{KEY}_{n\leftrightarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}] \overset{\pi_{\mathsf{pMAC}}}{\Longmapsto} \mathsf{A\text{-}AUT}_{n\to 1}.$$

  - In [BM20], Banfi and Maurer show that for a simple protocol π_{pE} based on key-indistinguishable and IND-CPA *probabilistic encryption* schemes,

$$[\mathsf{KEY}_{n\leftrightarrow 1}, \mathsf{A\text{-}AUT}_{n\to 1}] \overset{\pi_{\mathsf{pE}}}{\Longmapsto} \mathsf{A\text{-}SEC}_{n\to 1},$$

    and for a simple protocol π_{pAE} based on key-indistinguishable and IND-CCA3 *probabilistic authenticated encryption* schemes,

$$[\mathsf{KEY}_{n\leftrightarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}] \overset{\pi_{\mathsf{pAE}}}{\Longmapsto} \mathsf{A\text{-}SEC}_{n\to 1}.$$

- In the public-key setting, Kohlweiss et al. [KMO+13] show that for a simple protocol π_{PKE} based on key-indistinguishable and robust IND-CCA *public-key encryption* schemes,

$$[\mathsf{1\text{-}AUT}_{1\leftarrow n}, \mathsf{A\text{-}INS}_{1\to n}] \overset{\pi_{\mathsf{PKE}}}{\Longmapsto} \mathsf{A\text{-}CNF}_{1\to n}.$$

So far, no public-key constructions achieving sender anonymity were given, and we fill precisely this gap here, stated as an open problem in [KMO+13].

1.3 Contributions

Referring to the above discussion, it is natural to ask whether it is possible to construct $\mathsf{A\text{-}AUT}_{n\to 1}$ from $\mathsf{1\text{-}AUT}_{n\to 1}$ and $\mathsf{A\text{-}INS}_{n\to 1}$, using a protocol based on signature schemes achieving some form of anonymity. But it is rather easy to see that for regular signature schemes, this is impossible. Using an intuitive notation, the first result that we show is in fact that for any such protocol π,

$$[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{A\text{-}INS}_{n\to 1}] \overset{\pi}{\not\Longmapsto} \mathsf{A\text{-}AUT}_{n\to 1}, \qquad (1)$$

that is, no protocol can construct $\mathsf{A\text{-}AUT}_{n\to 1}$ from $\mathsf{1\text{-}AUT}_{n\to 1}$ and $\mathsf{A\text{-}INS}_{n\to 1}$ *only*. We prove this in Appendix B.

The main goal of this paper is to show how to get around this impossibility result by rethinking what can actually be achieved in this setting. We still did not discuss the guarantees of the receiver: In $\mathsf{A\text{-}AUT}_{n\to 1}$, while only the message m is leaked to the adversary, the receiver will see both the message m and the sender's identity S_i. Therefore, we identify two natural ways in which we can modify this resource such that we can then make meaningful statements. We see this systematic approach as a further contribution of this paper.

- We introduce the new resource *de-anonymizable authentic channel* $\mathsf{D\text{-}AUT}_{n\rightarrow 1}$, which is similar to $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$, except that it only guarantees authenticity of a sender once it decides to give up its anonymity. In more detail, a sender S_i can send a message m, and both the adversary and the receiver will only see m, but can decide at a later point to leak its identity to both parties, and this capability is not available to the adversary. This channel could be used for example in an anonymous auction, where bids need to be anonymous but the winner is required to later give up its anonymity in order to (authentically) claim the winning bet.
- We also introduce the new ideal resource *receiver-side anonymous authentic channel* $\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$, which is similar to $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$, except that the anonymity of the sender is guaranteed also towards the receiver, not just the adversary. Therefore, $\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$ also captures *internal* anonymity.

In the following table we summarize the guarantees provided by those resources.

Channel Name	**Symbol**	**Leaked**	**Inject**	**Received**
Anonymous & Authentic	$\mathsf{A\text{-}AUT}_{n\rightarrow 1}$	m	no	S_i, m
De-Anonymizable & Authentic	$\mathsf{D\text{-}AUT}_{n\rightarrow 1}$	$m/(S_i, m)$	$\tilde{m}/$~~$(S_j, \tilde{m})$~~	$m/(S_i, m)$
Receiver-Side Anon. & Authentic	$\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$	m	no	m

We can now summarize our contribution as providing constructions that, compared to (1), (i) use a different set of assumed resources, (ii) realize a different kind of ideal resource, or (iii) both. For (i) we show that a new scheme that we introduce, *bilateral signatures*, can be used to construct $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$ if we further assume a (single-use) authentic channel from the receiver to the senders, $\mathsf{1\text{-}AUT}_{n\leftarrow 1}$. Informally, we show that

$$[\mathsf{1\text{-}AUT}_{n\rightarrow 1}, \mathsf{1\text{-}AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\rightarrow 1}] \overset{\pi_{\mathsf{BS}}}{\Longmapsto} \mathsf{A\text{-}AUT}_{n\rightarrow 1},$$

which amounts to giving composable semantics to bilateral signatures. For (ii) we show that $\mathsf{D\text{-}AUT}_{n\rightarrow 1}$ can be constructed from the original set of assumed resources from (1) using *partial signatures* from [BD09,SY09]. Informally, we show that

$$[\mathsf{1\text{-}AUT}_{n\rightarrow 1}, \mathsf{A\text{-}INS}_{n\rightarrow 1}] \overset{\pi_{\mathsf{PS}}}{\Longmapsto} \mathsf{D\text{-}AUT}_{n\rightarrow 1},$$

which amounts to giving composable semantics to partial signatures. Finally, for (iii) we show that $\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$ can be constructed using *ring signatures* [RST01,BKM06] if instead of $\mathsf{1\text{-}AUT}_{n\rightarrow 1}$, we assume a (single-use) *broadcast authentic channel*, $\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}$, which from each sender authentically transmits a message to the receiver, as well as all other senders. Informally, we show that

$$[\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}, \mathsf{A\text{-}INS}_{n\rightarrow 1}] \overset{\pi_{\mathsf{RS}}}{\Longmapsto} \mathsf{RA\text{-}AUT}_{n\rightarrow 1},$$

which amounts to giving composable semantics to ring signatures.

1.4 Outline

In Section 2 we introduce our notation and the specific version of constructive cryptography used to present our results. We present and relate game-based and composable security notions for *bilateral signatures* in Section 3, for *partial signatures* in Section 4, and for *ring signatures* in Section 5.

2 Preliminaries

2.1 Notation

We write $x, \ldots \leftarrow y$ to assign the value y to variables $x, \ldots$, and $z, \ldots \leftarrow \mathcal{D}$ to assign independently and identically distributed values to variables $z, \ldots$ according to distribution $\mathcal{D}$, where we usually describe $\mathcal{D}$ as a probabilistic function. $\varnothing$ denotes the empty set, $\mathbb{N} \doteq \{0, 1, 2, \ldots\}$ denotes the set of natural numbers, and for $n \in \mathbb{N}$, we use the convention $[n] \doteq \{1, \ldots, n\}$. For a random variable X over a set $\mathcal{X}$, we define $\operatorname{supp} X \doteq \{x \in \mathcal{X} \mid \Pr[X = x] > 0\}$. For a logical statement S, $\mathbb{1}\{S\}$ is 1 if S is true, and 0 otherwise. Finally, for tuples we sometimes abuse notation in the following way: $(x, (y, z)) = (x, y, z)$.

2.2 Constructive Cryptography

In this work we use the composable framework of *constructive cryptography* (CC), originally introduced by Maurer and Renner [MR11,Mau12], incorporating ideas later exposed in [MR16] and [JM20]. At the most abstract level, CC is a theory that allows to define security of cryptographic protocols as statements about *constructions* transforming a number of resources satisfying some real (easier to achieve) *specification* $\mathcal{R}$ into a resource satisfying an ideal (simple and abstract) specification $\mathcal{S}$. In this work we use the version of CC in which a specification $\mathcal{S}$ is simply modeled as a subset of the set of all resources Φ, therefore, $\mathcal{S} \subseteq \Phi$. For a resource $\mathsf{R} \in \Phi$ we will often abuse notation and use the expression R in order to refer to the singleton specification $\{\mathsf{R}\}$.

On this abstract level, we define a *constructor* γ simply as a function $\Phi \to \Phi$, which given a resource $\mathsf{R} \in \Phi$, returns the constructed resource $\gamma(\mathsf{R}) \in \Phi$, and we also consider the natural lift-up $\gamma : 2^{\Phi} \to 2^{\Phi}$ of constructor γ to specifications by extending the definitions to include $\gamma(\mathcal{S}) \doteq \{\gamma(\mathsf{R}) \mid \mathsf{R} \in \mathcal{S}\} \subseteq \Phi$. Therefore, we formalize the concept of *construction* via the subset relation.

Definition 1. *Given specifications $\mathcal{R}, \mathcal{S} \subseteq \Phi$ and constructor $\gamma : \Phi \to \Phi$, γ* constructs *$\mathcal{R}$* from *$\mathcal{S}$, denoted $\mathcal{R} \xrightarrow{\gamma} \mathcal{S}$, if and only if $\gamma(\mathcal{R}) \subseteq \mathcal{S}$.*

Since this implies that $\mathcal{S}$, as a set, is potentially larger than $\gamma(\mathcal{R})$, it also highlights the fact that the guarantees given by the specification $\mathcal{S}$ are generally weaker than those given by $\mathcal{R}$. This results in $\mathcal{S}$ having simpler and easier to analyze guarantees, and therefore the statement can be interpreted as a distillation of the relevant properties.

Another important ingredient of CC is the concept of a *relaxation*. Given a resource $\mathsf{R} \in \Phi$, a relaxation $\rho : \Phi \to 2^{\Phi}$ maps R into a specification $\rho(\mathsf{R}) \subseteq \Phi$ and is such that $\mathsf{R} \in \rho(\mathsf{R})$. We use the shorthand notation $\mathsf{R}^{\rho} \doteq \rho(\mathsf{R})$. As we did for constructors, we also consider the natural lift-up $\rho : 2^{\Phi} \to 2^{\Phi}$ of a relaxation ρ to a specification $\mathcal{S} \subseteq \Phi$ by extending the definitions to include $\mathcal{S}^{\rho} \doteq \rho(\mathcal{S}) \doteq \bigcup_{\mathsf{R} \in \mathcal{S}} \mathsf{R}^{\rho} \subseteq \Phi$.

Systems, Resources, Converters, and Protocols. So far we defined CC on an abstract level, now we specify more concretely what kind of resources we consider, and how constructors are concretely instantiated for such objects. We model resources and constructors as random systems, just *systems* for short, as introduced in [Mau02] and later refined in [MPR07]. Simplistically, such mathematical objects can be considered as probabilistic discrete reactive systems, that can be queried with labeled inputs in a sequential fashion, where each distinct label corresponds to a distinct interface, and for each such input generate (possibly probabilistically) an equally labeled output depending on the input and the current state (formally defined by the sequence of all previous inputs and the associated outputs). Systems can be composed in parallel: given two (or more) systems $\mathbf{S}$ and $\mathbf{T}$, we denote $[\mathbf{S}, \mathbf{T}]$ as the system which can be independently queried at the interfaces of both $\mathbf{S}$ and $\mathbf{T}$. Following [BM20], we also use *correlated* parallel composition, where $\mathbf{S}$ and $\mathbf{T}$ are *not* independent (they might for example share a state, or depend on the same random variable), denoted $\langle \mathbf{S}, \mathbf{T} \rangle$. Such a system can be modeled by introducing another system $\mathbf{C}$ that has access to $[\mathbf{S}, \mathbf{T}]$, that is, $\langle \mathbf{S}, \mathbf{T} \rangle = \mathbf{C}[\mathbf{S}, \mathbf{T}]$.

In the following we consider only resources relevant to our setting for convenience, but of course everything can be phrased at a more abstract level for any kind of resource modeled as a system. Following [BM20], in this work all resources are parameterized by an integer $n \geq 2$, and each defines $n + 2$ interfaces: n for the senders, denoted S_i, for $i \in [n]$, one for the adversary, denoted E, one for the receiver, denoted R, and we define $\mathcal{I}_n \doteq \{S_1, \ldots, S_n, R, E\}$. In the following we use the expression n-resource to make explicit such parameter, and denote the set of all such resources as Φ_n. To any interface $I \in \mathcal{I}_n$ of an n-resource $\mathbf{R} \in \Phi_n$, we can attach a *converter* α (also formally modeled as a random system) which we assume results in a new n-resource, denoted as $\alpha^{I}\, \mathbf{R} \in \Phi_n$. We denote the set of all converters as Σ, and assume that they naturally compose, that is, for converters $\alpha, \beta \in \Sigma$, $\alpha\beta \in \Sigma$ is also a converter. Moreover, we assume commutativity of converters attached at different interfaces, that is, considering converters $\alpha, \beta \in \Sigma$ and interfaces $I, J \in \mathcal{I}_n$, with $I \neq J$, then $\alpha^{I}\, \beta^{J}\, \mathbf{R} = \beta^{J}\, \alpha^{I}\, \mathbf{R}$. Finally, we define the special converter $id \in \Sigma$ as the *identity converter* such that $id^{I}\, \mathbf{R} = \mathbf{R}$, for any $\mathbf{R} \in \Phi_n$ and $I \in \mathcal{I}_n$.

In order to make security statements using CC, we still need to define constructors for this specific type of resources. To do so, we first model a protocol π executed by n senders and one receiver (an n-protocol) as a list of $n + 1$ converters $(\alpha_1, \ldots, \alpha_{n+1})$, where the adopted convention is that α_i is attached to sender interface S_i, for $i \in [n]$, while α_{n+1} is attached to the receiver interface

R. In the following, we use the short-hand notation $\pi\mathbf{R}$ for the n-resource $\alpha_1^{S_1} \cdots \alpha_n^{S_n}\, \alpha_{n+1}^{R}\, \mathbf{R}$. This way, we can now instantiate the concept of a constructor γ simply as attachment of a n-protocol, that is, for each n-protocol π, we consider the associated constructor γ_π, and define $\gamma_\pi(\mathbf{R}) \doteq \pi\mathbf{R}$. Moreover, for a second n-protocol $\pi' \doteq (\beta_1, \ldots, \beta_{n+1})$, we define the composition of π' with π as $\pi'\pi \doteq (\beta_1\alpha_1, \ldots, \beta_{n+1}\alpha_{n+1})$, and therefore $\pi'\pi\mathbf{R}$ is the n-resource $(\beta_1\alpha_1)^{S_1} \cdots (\beta_n\alpha_n)^{S_n}\, (\beta_{n+1}\alpha_{n+1})^{R}\, \mathbf{R}$. Therefore composition of the constructors corresponding to π and π', that is, $\gamma_{\pi'} \circ \gamma_\pi$, is simply modeled as $\pi'\pi$. In the following, we will just use the concept of protocol attachment rather than the more abstract concept of a constructor.

Finally, for n-resources $\mathbf{R}_1, \ldots, \mathbf{R}_\ell \in \Phi_n$, we overload notation and define their parallel composition $[\mathbf{R}_1, \ldots, \mathbf{R}_\ell]$ also as an n-resource, but where each interface $I \in \mathcal{I}_n$ exports ℓ sub-interfaces I_j, for $j \in [\ell]$, with the convention that I_j provides direct access to the interface I of $\mathbf{R}_j$.

Indistinguishability of Systems. In order to define security, we also need to formalize the notion of indistinguishability of n-resources, and more in general of systems. For that, we formally define a *distinguisher* $\mathbf{D}$, also as a system but with the exception that it initially produces an output with no need for an input, and finally produces a binary output (which depends on the probabilistic interaction with another system). We always tacitly assume that a distinguisher $\mathbf{D}$ interacting with any system $\mathbf{S}$ has *matching interfaces* with $\mathbf{S}$; for n-resources we denote the set of all such distinguishers as Θ_n. We can attach a converter $\alpha \in \Sigma$ also to any distinguisher $\mathbf{D}$, at any of its interfaces, say I, which we assume results in a new distinguisher, denoted as $\mathbf{D}^{\,I}\alpha$, and in the case of an n-protocol π (and $\mathbf{D}$ being an appropriate distinguisher for an n-resource) we can naturally consider $\mathbf{D}\pi$. For a distinguisher $\mathbf{D}$ and systems $\mathbf{S}, \mathbf{T}$, we denote $\mathbf{D}$'s output after interacting with $\mathbf{S}$ as $\mathbf{DS} \in \{0, 1\}$, and define $\mathbf{D}$'s advantage in $\mathbf{S}$ from $\mathbf{T}$ as

$$\Delta^{\mathbf{D}}(\mathbf{S}, \mathbf{T}) \doteq |\Pr[\mathbf{DS} = 0] - \Pr[\mathbf{DT} = 0]|.$$

Considering a converter $\alpha \in \Sigma$ and an interface I, note that $\mathbf{D}^{\,I}\alpha\mathbf{S} = \mathbf{D}\alpha^{I}\,\mathbf{S}$, and therefore $\Delta^{\mathbf{D}}(\alpha^{I}\,\mathbf{S}, \alpha^{I}\,\mathbf{T}) = \Delta^{\mathbf{D}^{\,I}\alpha}(\mathbf{S}, \mathbf{T})$. Finally, given a function ε that maps distinguishers to $[0, 1]$, we can define the ε-indistinguishability relation between systems $\mathbf{S}$ and $\mathbf{T}$, called *ε-closeness*, as

$$\mathbf{S} \approx_\varepsilon \mathbf{T} \quad :\Longleftrightarrow \quad \forall \mathbf{D} :\ \Delta^{\mathbf{D}}(\mathbf{S}, \mathbf{T}) \leq \varepsilon(\mathbf{D}).$$

For a distinguisher $\mathbf{D}$, $\varepsilon(\mathbf{D})$ might be a negligible value (depending on some security parameter, which we do not make explicit in this work). More generally, ε maps a distinguisher $\mathbf{D}$ for systems $\mathbf{S}$ and $\mathbf{T}$ to the advantage that a new distinguisher $\tilde{\mathbf{D}}$ has in distinguishing two different systems $\tilde{\mathbf{S}}$ and $\tilde{\mathbf{T}}$, where $\tilde{\mathbf{D}}$ uses $\mathbf{D}$ as a black-box. For this we need to define a *reduction* system $\mathbf{C}$ that on one side exports all the interfaces of $\mathbf{D}$ (which has the same interface set as $\mathbf{S}$ and $\mathbf{T}$), and on the other side exports all interfaces of $\tilde{\mathbf{S}}$ (which has the same interface set as $\tilde{\mathbf{T}}$). Then if $\mathbf{C}$ is composed with $\tilde{\mathbf{S}}$ or $\tilde{\mathbf{T}}$, denoted $\mathbf{C}\tilde{\mathbf{S}}$ or $\mathbf{C}\tilde{\mathbf{T}}$, respectively, we usually show that $\mathbf{S} = \mathbf{C}\tilde{\mathbf{S}}$ and $\mathbf{T} = \mathbf{C}\tilde{\mathbf{T}}$. Just as we did for converters, we can assume more generally that such a system $\mathbf{C}$ can be attached to $\mathbf{D}$ resulting in a distinguisher system $\tilde{\mathbf{D}} \doteq \mathbf{DC}$ for $\tilde{\mathbf{S}}$ and $\tilde{\mathbf{T}}$. Then if we know (or assume) that $\tilde{\mathbf{S}} \approx_{\tilde{\varepsilon}} \tilde{\mathbf{T}}$, we have

$$\varepsilon(\mathbf{D}) = \Delta^{\mathbf{D}}(\mathbf{S}, \mathbf{T}) = \Delta^{\mathbf{D}}(\mathbf{C}\tilde{\mathbf{S}}, \mathbf{C}\tilde{\mathbf{T}}) = \Delta^{\mathbf{DC}}(\tilde{\mathbf{S}}, \tilde{\mathbf{T}}) \leq \tilde{\varepsilon}(\mathbf{DC}),$$

and by defining $\tilde{\varepsilon}^{\mathbf{C}}(\mathbf{D}) \doteq \tilde{\varepsilon}(\mathbf{DC})$, we establish the (function) inequality $\varepsilon \leq \tilde{\varepsilon}^{\mathbf{C}}$ which entails that by showing (or just assuming) that $\tilde{\varepsilon}$ is negligible (for all distinguishers), then so is ε.

Relevant Resource Specification Relaxations. Recall that for our specific instantiation of CC, a specification $\mathcal{S} \subseteq \Phi_n$ is a set of n-resources. Then for a converter $\alpha \in \Sigma$ and an interface $I \in \mathcal{I}_n$, we define $\alpha^I \mathcal{S} \doteq \{\alpha^I \mathbf{R} \mid \mathbf{R} \in \mathcal{S}\}$, and for an n-protocol π, we analogously define $\pi\mathcal{S} \doteq \{\pi\mathbf{R} \mid \mathbf{R} \in \mathcal{S}\}$. Next we define two important relaxations, as introduced in [MR16].

First, we define the *ε-relaxation* of $\mathbf{R}$ as the set of all resources which are ε-close to $\mathbf{R}$, for some function $\varepsilon : \Theta_n \to [0, 1]$, that is,

$$\mathbf{R}^{\varepsilon} \doteq \{\mathbf{S} \in \Phi_n \mid \mathbf{S} \approx_{\varepsilon} \mathbf{R}\}.$$

We can naturally extend this notion to a specification $\mathcal{S} \subseteq \Phi_n$, that is, we define

$$\mathcal{S}^{\varepsilon} \doteq \bigcup_{\mathbf{R} \in \mathcal{S}} \mathbf{R}^{\varepsilon} = \{\mathbf{S} \in \Phi_n \mid \exists \mathbf{R} \in \mathcal{S} : \mathbf{S} \approx_{\varepsilon} \mathbf{R}\}.$$

Secondly, we define the *$*$-relaxation* (spelled "star relaxation") of $\mathbf{R}$, relative to a set of interfaces $\mathcal{C} \subseteq \mathcal{I}_n$, with $t \doteq |\mathcal{C}|$ and $\mathcal{C} \doteq \{I_1, \ldots, I_t\}$, as the set of all resources which behave arbitrarily at those interfaces, that is,

$$\mathbf{R}^{*_{\mathcal{C}}} \doteq \{\alpha_1^{I_1} \cdots \alpha_t^{I_t} \mathbf{R} \mid \alpha_1, \ldots, \alpha_t \in \Sigma\}.$$

We can again extend this notion to a specification $\mathcal{S} \subseteq \Phi_n$, that is, we define

$$\mathcal{S}^{*_{\mathcal{C}}} \doteq \bigcup_{\mathbf{R} \in \mathcal{S}} \mathbf{R}^{*_{\mathcal{C}}} = \{\alpha_1^{I_1} \cdots \alpha_t^{I_t} \mathbf{R} \mid \alpha_1, \ldots, \alpha_t \in \Sigma, \mathbf{R} \in \mathcal{S}\}.$$

This relaxation intuitively captures a scenario in which a set of parties is dishonest, namely those which are assigned to the interfaces in $\mathcal{C}$. We often consider the singleton $\mathcal{C} = \{E\}$ for which we write $\mathbf{R}^{*_E}$ and $\mathcal{S}^{*_E}$ instead of $\mathbf{R}^{*_{\{E\}}}$ and $\mathcal{S}^{*_{\{E\}}}$, respectively.

Constructions Capturing Anonymity. Using the specifications introduced above, we can now illustrate the specific type of construction statements that we will show in this work. Intuitively, we want to say that a weaker (that is, "smaller") specification $\mathcal{S}$ can be constructed from a stronger (that is, "larger") specification $\mathcal{R}$ by an n-protocol π if applying π to any n-resource $\mathbf{R} \in \mathcal{R}$ satisfying the specification $\mathcal{R}$, results in an n-resource $\pi\mathbf{R} \in \Phi_n$ not too far from an n-resource $\mathbf{S} \in \mathcal{S}$ satisfying the specification $\mathcal{S}$. As usual in cryptography, we also require that such n-resource $\mathbf{S}$ can exhibit arbitrary behavior at the adversarial interface E, reflecting the fact that whatever the adversary can do in the real-world, modeled by $\pi\mathbf{R}$, it can also do in the ideal-world. This is conventionally modeled by considering a special converter $\mathsf{sim} \in \Sigma$ (a *simulator*) that is attached to $\mathbf{S}$'s adversarial interface E, resulting in the resource $\mathsf{sim}^E\,\mathbf{S} \in \Phi_n$. Therefore, on a high level the statement that one need to prove is $\mathcal{R} \xrightarrow{\pi} \mathcal{S}^{*_E}$, if we would consider perfect closeness, that is, information theoretic security. But more in general, we formalize the concept of "not too far" by means of the ε-relaxation, hence a more frequent kind of statement to prove in cryptography is $\mathcal{R} \xrightarrow{\pi} (\mathcal{S}^{*_E})^{\varepsilon}$, which essentially allows us to rely on cryptographic assumptions.

But this specific type of construction still does not allow us to appropriately model anonymity in our setting; in order to capture anonymity, we exploit the power of the $*$-relaxation once more. Concretely, as pointed out earlier, we want to make statements about the *preservation* of anonymity: We want to capture that a protocol neither increases, nor degrades anonymity, and we do so by modeling the "level" of anonymity by a corruption set $\mathcal{C} \subseteq \{S_i\}_{i=1}^n$. Then we show that for any such corruption set $\mathcal{C}$, if the senders which are *not* part of such set execute the protocol, they still obtain the desired properties. To formalize this for a protocol $\pi \doteq (\alpha_1, \ldots, \alpha_{n+1})$, we use the notation $\pi^{\overline{\mathcal{C}}}$, by which we mean the list of protocols $(\alpha_1, \ldots, \alpha_{n+1})$, but where for any $S_i \in \mathcal{C}$, for some $i \in [n]$, α_i is replaced by the identity converter *id*. We now formalize the specific type of construction statements that we will make (and in Appendix C we show they compose).

Definition 2. *For an n-protocol π, a function ε, and n-resources $\mathbf{R}, \mathbf{S}$,* π anonymously constructs $\mathbf{S}$ from $\mathbf{R}$ within ε, *denoted* $\mathbf{R} \overset{\pi,\varepsilon}{\Longmapsto} \mathbf{S}$, *if for all $\mathcal{C} \subseteq \{S_i\}_{i=1}^n$, $\mathbf{R}^{*_\mathcal{C}} \xrightarrow{\pi^{\overline{\mathcal{C}}}} (\mathbf{S}^{*_{\mathcal{C}\cup\{E\}}})^{\varepsilon}$, that is, $\pi^{\overline{\mathcal{C}}}\,\mathbf{R}^{*_\mathcal{C}} \subseteq (\mathbf{S}^{*_{\mathcal{C}\cup\{E\}}})^{\varepsilon}$.*

2.3 Anonymous and Authentic Resources

In this section we present the n-resources that we need later in order to make our security statements, and we formally define then all in Appendix A. Instead of bold-face letters, for such resources we will use suggestive sans-serif abbreviations. We describe all resources first on an intuitive level, and then formally following the model introduced in [BM20], in which communication is modeled by a sender buffer $\mathfrak{S}$ and a receiver buffer $\mathfrak{R}$, both allowing to insert single elements and to read in chunks. Note that all our resources are parameterized by a set, either $\mathcal{K}$ (ideally for public keys), $\mathcal{M}$ (ideally for messages), or $\mathcal{X}$ (for anything), but we will make the instantiation of such set implicit when showing constructions.

We begin by describing the three single-use authentic channels needed as assumed resources in order to authentically exchange public keys. The first such resource is $\mathsf{1\text{-}AUT}_{n\to 1}$, which allows to input a value once at every sender interface S_i, for $i \in [n]$, and allows to read these values at the receiver and adversary interfaces, R and E, respectively. Based on this resource, we then simply define $\mathsf{1\text{-}AUT}_{n\leftarrow 1}$ as somewhat the dual of this, namely, the resource that allows to input a value once at the receiver interface R, and that allows to read this value at every sender and adversary interface, S_i, for $i \in [n]$, and E, respectively. Finally, we also need the resource $\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}$, which similarly to $\mathsf{1\text{-}AUT}_{n\rightarrow 1}$ allows to input a value once at every sender interface S_i, for $i \in [n]$, but additionally allows to read these values at all the sender interfaces S_i as well. We tacitly assume that protocols first use those resources to exchange public-keys, and only once all keys have been exchanged, they use the channel resources. We also point out that our results are in a model in which public keys are therefore assumed to always be honestly generated. We leave open the problem of strengthening the model by replacing these resources by a *certificate authority*, which would allow the adversary to also register keys.

We next describe the assumed channel resource $\mathsf{A\text{-}INS}_{n\rightarrow 1}$ as well as the three different ideal anonymous channel resources $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$, $\mathsf{D\text{-}AUT}_{n\rightarrow 1}$, and $\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$ (all depicted in Figure 1).

- The *anonymous insecure channel* $\mathsf{A\text{-}INS}_{n\rightarrow 1}$ allows to input multiple values at every sender interface S_i, for $i \in [n]$. Those values are stored in the sender buffer $\mathfrak{S}$, from which they can be read at the adversary interface E. Moreover, at this interface $\mathsf{A\text{-}INS}_{n\rightarrow 1}$ also allows the adversary to inject multiple arbitrary values. Those values are stored in the receiver buffer $\mathfrak{R}$, from which they can be read at the receiver interface R.
- In the *anonymous authentic channel* $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$, the sender buffer $\mathfrak{S}$ is used exactly as in $\mathsf{A\text{-}INS}_{n\rightarrow 1}$, except that for every message sent, information about the sender is also stored, but not leaked to the adversary. Unlike $\mathsf{A\text{-}INS}_{n\rightarrow 1}$, at the interface E, $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$ only allows the adversary to select which messages previously input by a sender will be transmitted to the receiver. Those messages, along with the sender information, will be transferred from the sender buffer $\mathfrak{S}$ to the receiver buffer $\mathfrak{R}$, from which they can be read at the receiver interface R.
- The *de-anonymizable authentic channel* $\mathsf{D\text{-}AUT}_{n\rightarrow 1}$ allows to input two type of values at every sender interface S_i, for $i \in [n]$: one to commit a message m, $(\underline{\texttt{cmt}}, m)$, and the other to authenticate a previously committed message m', $(\underline{\texttt{aut}}, h_{m'})$, where $h_{m'}$ is a handle for m' generated by $\mathsf{D\text{-}AUT}_{n\rightarrow 1}$. Those values are stored in the sender buffer $\mathfrak{S}$, from which they can be read at the adversary interface E. Information about the sender is also stored, but is only leaked to the adversary along with $\underline{\texttt{aut}}$ values. At the interface E, $\mathsf{D\text{-}AUT}_{n\rightarrow 1}$ allows the adversary to select which values (of both types) previously input by a sender will be transmitted to the receiver, as well as to inject additional $\underline{\texttt{cmt}}$ values. Those values, including sender information only in case of $\underline{\texttt{aut}}$ values, will be transferred from the sender buffer $\mathfrak{S}$ to the receiver buffer $\mathfrak{R}$, from which they can be read at the receiver interface R.
- The *receiver-side anonymous authentic channel* $\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$ works exactly as $\mathsf{A\text{-}AUT}_{n\rightarrow 1}$, except that sender information is concealed from the receiver as well (and therefore never stored in the buffers $\mathfrak{S}$ and $\mathfrak{R}$).

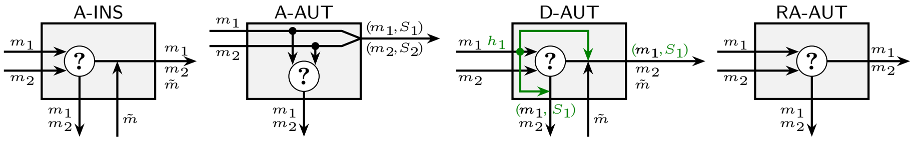

Fig. 1. Sketches of the anonymous channel resources for $n = 2$ senders (S_1 sending m_1 and S_2 sending m_2). For $\mathsf{D\text{-}AUT}$, only S_1 de-anonymizes its message (in green).

3 Achieving Anonymous Authenticity

We start by introducing a new flavor of a signature scheme with some anonymity property, dubbed *bilateral signatures*. This scheme shares the syntax of designated verifier signatures (DVS): both sender and receiver have a key-pair; signing a message requires the secret key of the sender and the public key of the receiver, and verifying a signature requires the secret key of the receiver and the public key of the sender. The receiver's key-pair is essentially what allows to circumvent the impossibility result from Appendix B, by introducing one-time authenticated information from the receiver to the senders: it enables indistinguishability of signatures by making verification exclusive to the receiver, as opposed to public.

Definition 3 (Bilateral Signature Scheme). *A* bilateral signature scheme (BSS) $\Sigma_{\mathsf{BS}} \doteq (\mathtt{Gen}_S, \mathtt{Gen}_R, \mathtt{Sgn}, \mathtt{Vrf})$ *over message-space $\mathcal{M}$ and signature-space $\mathcal{S}$ (with $\bot \notin \mathcal{M} \cup \mathcal{S}$), is such that*

- $\mathtt{Gen}_S$ *is a distribution over the sender key-spaces $\mathcal{SK}_S \times \mathcal{PK}_S$;*
- $\mathtt{Gen}_R$ *is a distribution over the receiver key-spaces $\mathcal{SK}_R \times \mathcal{PK}_R$;*
- $\mathtt{Sgn} : \mathcal{SK}_S \times \mathcal{PK}_R \times \mathcal{M} \to \mathcal{S}$ *is a probabilistic function;*
- $\mathtt{Vrf} : \mathcal{SK}_R \times \mathcal{PK}_S \times \mathcal{M} \times \mathcal{S} \to \{0, 1\}$ *is a deterministic function.*

We require the above to be efficiently samplable/computable. For sender key-pair $(ssk, spk) \in \mathcal{SK}_S \times \mathcal{PK}_S$ and receiver key-pair $(rsk, rpk) \in \mathcal{SK}_R \times \mathcal{PK}_R$ we use the short-hand notation $\mathtt{Sgn}_{ssk,rpk}(\cdot)$ for $\mathtt{Sgn}(ssk, rpk, \cdot)$ and $\mathtt{Vrf}_{rsk,spk}(\cdot, \cdot)$ for $\mathtt{Vrf}(rsk, spk, \cdot, \cdot)$. Moreover, we assume correctness *of Σ_{BS}, that is, for all key-pairs (ssk, spk) and (rsk, rpk) distributed according to $\mathtt{Gen}_S$ and $\mathtt{Gen}_R$, respectively, all messages $m \in \mathcal{M}$, and all signatures $\sigma \in \mathcal{S}$, $\mathtt{Vrf}_{rsk,spk}(m, \sigma) = \mathbb{1}\{\sigma \in \operatorname{supp}(\mathtt{Sgn}_{ssk,rpk}(m))\}$.*

Note that we only introduce bilateral signatures as an abstract syntactic object. As we discuss in Appendix E.1, there exist concrete schemes satisfying such syntax, as well as the semantics we define later. Nevertheless, such schemes provide additional security guarantees that are not required in our setting. We leave the problem of finding a bilateral signature scheme which is *minimal.*

3.1 Game-Based Security of Bilateral Signatures

We begin our study of the semantics of bilateral signatures by defining their game-base security. In order to define the security of a fixed scheme Σ_{BS}, we define the following systems (where the dependency on Σ_{BS} is implicit), parameterized by keys $(ssk, spk) \in \mathcal{SK}_S \times \mathcal{PK}_S$, $\boldsymbol{spk} \doteq (spk_1, \ldots, spk_n) \in \mathcal{PK}_S^n$, for any $n \in \mathbb{N}$, and $(rsk, rpk) \in \mathcal{SK}_R \times \mathcal{PK}_R$.

- $\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}_{rsk,spk} \rangle$:
  - On input $m \in \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $\sigma \leftarrow \mathtt{Sgn}_{ssk,rpk}(m)$.
  - On input $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, return m if $\mathtt{Vrf}_{rsk,spk}(m, \sigma) = 1$ and $\perp$ otherwise.
- $\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}^{\perp} \rangle$: Set $\mathcal{Q} \subseteq \mathcal{M} \times \mathcal{S}$ to $\varnothing$ and then:
  - On input $m \in \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $\sigma \leftarrow \mathtt{Sgn}_{ssk,rpk}(m)$, and set $\mathcal{Q}$ to $\mathcal{Q} \cup \{(m, \sigma)\}$.
  - On input $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, return m if $(m, \sigma) \in \mathcal{Q}$ and $\perp$ otherwise.
- $\mathbf{K}_{\boldsymbol{spk},rpk}$: On input $\diamond$, output $(\boldsymbol{spk}, rpk)$.

In our definitions, all keys will *always* be random variables distributed (as key-pairs) according to Σ_{BS}'s $\mathtt{Gen}_S$ and $\mathtt{Gen}_R$.

We define a combined notion for bilateral signatures capturing both authenticity and anonymity at once. For this, we define a distinction problem between a real system that correctly generates and verifies signatures, via signing and verification oracles for n (different) senders and one receiver, and an ideal system that also correctly generates signatures and only correctly verifies signatures previously signed, but via n copies of signing and verification oracles for the *same* (randomly selected) sender and one receiver.

Definition 4 (UF-IK-Secure Bilateral Signature). *A bilateral signature scheme Σ_{BS} is (n, ε)-*unforgeable-and-anonymous *(or (n, ε)-*UF-IK-secure*) if*

$$[\langle \mathbf{S}_{ssk_1,rpk}, \mathbf{V}_{rsk,spk_1} \rangle, \ldots, \langle \mathbf{S}_{ssk_n,rpk}, \mathbf{V}_{rsk,spk_n} \rangle, \mathbf{K}_{\boldsymbol{spk},rpk}]$$
$$\approx_{\varepsilon}$$
$$[\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}]$$

for key-pairs $(ssk_1, spk_1), \ldots, (ssk_n, spk_n) \leftarrow \mathtt{Gen}_S$, $(rsk, rpk) \leftarrow \mathtt{Gen}_R$, $\boldsymbol{spk} \doteq (spk_1, \ldots, spk_n)$, and random variable $I \overset{\$}{\leftarrow} [n]$.

As we formally show in Appendix D.1, it is easy to see that if a bilateral signature scheme is ε-UF-secure *and* (n, ε')-IK-secure (as defined there), then it is $(n, \varepsilon^{\mathbf{C}} + \varepsilon')$-UF-IK-secure, for a specific reduction $\mathbf{C}$.

3.2 Composable Security of Bilateral Signatures

We continue our study of the semantics of bilateral signatures by defining their composable security in the constructive cryptography framework. Recall that we want to define composable security of a bilateral signature scheme Σ_{BS} as the construction of the resource $\mathsf{A\text{-}AUT}_{n \to 1}$ from the resources $\mathsf{1\text{-}AUT}_{n \to 1}$, $\mathsf{1\text{-}AUT}_{n \leftarrow 1}$, and $\mathsf{A\text{-}INS}_{n \to 1}$ (instantiated with $\mathcal{X} = \mathcal{M} \times \mathcal{S}$, referring to Appendix A). In order to make this statement formal, we need to define how a protocol π_{BS}, attached to the resource $[\textsf{1-AUT}_{n\to 1}, \textsf{1-AUT}_{n\leftarrow 1}, \textsf{A-INS}_{n\to 1}]$, naturally makes use of $\Sigma_{\textsf{BS}}$. First, $\pi_{\textsf{BS}}$ runs $\texttt{Gen}_S$ for every sender S_i, for $i \in [n]$, generating key-pairs $(ssk_1, spk_1), \ldots, (ssk_n, spk_n)$, as well as $\texttt{Gen}_R$ for the receiver R, generating the key-pair (rsk, rpk). Then it transmits the sender public keys $spk_1, \ldots, spk_n$ to the receiver through $\textsf{1-AUT}_{n\to 1}$ and the receiver public key rpk to each of the senders through $\textsf{1-AUT}_{n\leftarrow 1}$. After that, once a sender S_i inputs a message m on its interface, $\pi_{\textsf{BS}}$ uses ssk_i and rpk to generate $\sigma \leftarrow \texttt{Sgn}_{ssk_i, rpk}(m)$, and inputs (m, σ) to the interface S_i of $\textsf{A-INS}_{n\to 1}$. Once the receiver R inputs $\diamond$ on its interface, $\pi_{\textsf{BS}}$ also inputs $\diamond$ to the interface R of $\textsf{A-INS}_{n\to 1}$, obtaining a set $\mathfrak{O} \subseteq \mathbb{N} \times \mathcal{M} \times \mathcal{S}$, and outputs the set $\{(j, m, i) \mid \exists\,(j, m, \sigma) \in \mathfrak{O}, i \in [n] : \texttt{Vrf}_{rsk, spk_i}(m, \sigma) = 1\}$ to R. We call $\pi_{\textsf{BS}}$ the protocol using $\Sigma_{\textsf{BS}}$ in the *natural way.*

Definition 5. *A bilateral signature scheme* $\Sigma_{\textsf{BS}}$ *is* (n, ε)-composably secure *if*

$$[\textsf{1-AUT}_{n\to 1}, \textsf{1-AUT}_{n\leftarrow 1}, \textsf{A-INS}_{n\to 1}] \xmapsto{\pi_{\textsf{BS}}, \varepsilon} \textsf{A-AUT}_{n\to 1},$$

where $\pi_{\textsf{BS}}$ *is the protocol using* $\Sigma_{\textsf{BS}}$ *in the natural way.*

Finally, we show that game-based security of bilateral signatures implies their composable security (we defer the proof to Appendix F.1).

Theorem 1. *There exists a reduction system* $\mathbf{C}$ *such that, if a bilateral signature scheme* $\Sigma_{\textsf{BS}}$ *is* (n, ε)-$\textsf{UF-IK}$*-secure, then it is* $(n, \varepsilon^{\mathbf{C}})$*-composably secure.*

4 Achieving De-Anonymizable Authenticity

In the previous section we studied a way to achieve the anonymous resource $\textsf{A-AUT}_{n\to 1}$, at the cost of assuming additional one-time authenticated information from the receiver to all senders. In this section we tackle what can be interpreted as the dual problem, that is, we study what can at most be achieved by only assuming one-time authenticated information from the receivers to the sender (in addition to an insecure channel). Considering to our impossibility result from Appendix B, we know that the constructed resource will need to be weaker than $\textsf{A-AUT}_{n\to 1}$.

Considering the constraint on the assumed resources, intuitively we need a scheme that, on the sender side, requires the same input as regular signatures, that is, just a secret key and a message. But since anonymity is unachievable if both the message and the signature are disclosed, one either needs to relax the security definition of digital signatures, or to slightly change their syntax.

A first workaround to this impossibility was initially studied by Yang et al. [YWDW06], and subsequently refined independently by Fischlin [Fis07] and Zhang and Imai [ZI09], where the first approach is taken and essentially the anonymity of the signature alone is considered. Modeling such a security definition composably, makes it apparent how, from an application point of view, this approach is moot: it requires to assume that an adversary only sees signatures in transit, but not messages. Clearly, a different kind of assumed resources is needed; ideally, the message should be transmitted over a confidential channel. Composably, this hints to the fact that anonymous signatures might only be appropriate in a context where one wants to combine signatures with public-key encryption. This can be interpreted as the study of anonymity preservation of signcryption, and we briefly discuss this in Appendix G.

A different workaround, following the second approach, was independently taken later by Saraswat and Yun [SY09] and by Bellare and Duan [BD09]. There, the syntax of regular DSS was slightly modified to allow the signature to bear some form of anonymity. More precisely, the security definitions are changed to capture anonymity when the message and only a portion of the signature are disclosed, and authenticity only once the full signature is disclosed. We remark that the two works essentially introduce the same syntax and security notions, but [SY09] uses the term anonymous signatures introduced earlier in [YWDW06], whereas [BD09] adopts the new term *partial signatures*, which we will adopt here as well. More precisely, in such a scheme the signing function returns a signature that is defined as a tuple (σ, τ), where σ is called the *stub*, τ the *tag*, and (σ, τ) the *full signature*. Then the stub σ alone guarantees anonymity of the sender on a message m (but not its authenticity), whereas authenticity of m (but not anonymity anymore) is guaranteed once the tag τ is subsequently disclosed.

Definition 6 (Partial Signature Scheme). *A* partial signature scheme (PSS) $\Sigma_{\mathsf{PS}} \doteq (\mathtt{Gen}, \mathtt{Sgn}, \mathtt{Vrf})$ *over message-space* $\mathcal{M}$*, stub-space* $\mathcal{S}$*, and tag-space* $\mathcal{T}$ *(with* $\perp \notin \mathcal{M} \cup \mathcal{S} \cup \mathcal{T}$*), is such that*

- $\mathtt{Gen}$ *is a distribution over the key-spaces* $\mathcal{SK} \times \mathcal{PK}$*;*
- $\mathtt{Sgn} : \mathcal{SK} \times \mathcal{M} \to \mathcal{S} \times \mathcal{T}$ *is a probabilistic function;*
- $\mathtt{Vrf} : \mathcal{PK} \times \mathcal{M} \times \mathcal{S} \times \mathcal{T} \to \{0, 1\}$ *is a deterministic function.*

We require the above to be efficiently samplable/computable. For key-pair $(sk, pk) \in \mathcal{SK} \times \mathcal{PK}$ *we use the short-hand notation* $\mathtt{Sgn}_{sk}(\cdot)$ *for* $\mathtt{Sgn}(sk, \cdot)$ *and* $\mathtt{Vrf}_{pk}(\cdot, \cdot, \cdot)$ *for* $\mathtt{Vrf}(pk, \cdot, \cdot, \cdot)$*. Moreover, we assume* correctness *of* Σ_{PS}*, that is, for all key-pairs* (sk, pk) *distributed according to* $\mathtt{Gen}$*, all messages* $m \in \mathcal{M}$*, and all signatures* $(\sigma, \tau) \in \mathcal{S} \times \mathcal{T}$*,* $\mathtt{Vrf}_{pk}(m, \sigma, \tau) = \mathbb{1}\{(\sigma, \tau) \in \operatorname{supp}(\mathtt{Sgn}_{sk}(m))\}$*.*

4.1 Game-Based Security of Partial Signatures

We begin our study of the semantics of partial signatures by defining their game-base security. Originally, in [YWDW06] anonymous signatures (the precursors of partial signatures), were only defined to be unforgeable and anonymous, by requiring that no adversary can forge valid signatures and distinguish signatures when messages are withheld, respectively. In [SY09] and [BD09], for the succeeding partial signatures, the unforgeability notion is essentially unchanged, whereas anonymity is defined with a game where the adversary sees only a part of the signatures, but also the whole associated messages. Additionally, both works realize that a crucial third security guarantee is also necessary: *unambiguouity* (named unpretendability in [SY09]). This notion ensures that only the original creator of a signature is able to later show that it indeed generated it. This security guarantee is modeled via a game where an adversary tries to come up with two messages m_0, m_1, a stub σ, and two tags τ_0, τ_1, such that $\mathtt{Vrf}_{pk_0}(m_0, \sigma, \tau_0) = \mathtt{Vrf}_{pk_1}(m_1, \sigma, \tau_1) = 1$, for two different public keys pk_0, pk_1, which in our setting must be two of the n known (and fixed) sender public keys. In Appendices D.2 and E.2 we relate those notions from the literature to the new definitions we introduce next.

In order to define the security of a fixed scheme Σ_{PS}, we define the following systems (where the dependency on Σ_{PS} is implicit), parameterized by keys $sk \in \mathcal{SK}$, $pk \in \mathcal{PK}$, $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n) \in \mathcal{PK}^n$, for any $n \in \mathbb{N}$.

- $\langle \mathbf{S}_{sk}, \mathbf{V}_{pk} \rangle$:
  - On input $m \in \mathcal{M}$, return $(m, \sigma, \tau) \in \mathcal{M} \times \mathcal{S} \times \mathcal{T}$, for $(\sigma, \tau) \leftarrow \mathtt{Sgn}_{sk}(m)$.
  - On input $(m, \sigma, \tau) \in \mathcal{M} \times \mathcal{S} \times \mathcal{T}$, return m if $\mathtt{Vrf}_{pk}(m, \sigma, \tau) = 1$ and $\perp$ otherwise.
- $\langle \mathbf{S}_{sk}, \mathbf{V}^{\perp} \rangle$: Set the (potentially) *shared* set $\mathcal{Q} \subseteq \mathcal{M} \times \mathcal{S} \times \mathcal{T}$ to $\varnothing$ and then:
  - On input $m \in \mathcal{M}$, return $(m, \sigma, \tau) \in \mathcal{M} \times \mathcal{S} \times \mathcal{T}$, for $(\sigma, \tau) \leftarrow \mathtt{Sgn}_{sk}(m)$, and set $\mathcal{Q}$ to $\mathcal{Q} \cup \{(m, \sigma, \tau)\}$.
  - On input $(m, \sigma, \tau) \in \mathcal{M} \times \mathcal{S} \times \mathcal{T}$, return m if $(m, \sigma, \tau) \in \mathcal{Q}$ and $\perp$ otherwise.
- $\mathbf{S}^{-}_{sk}$: On input $m \in \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $(\sigma, \cdot) \leftarrow \mathtt{Sgn}_{sk}(m)$.
- $\mathbf{K}_{\boldsymbol{pk}}$: On input $\diamond$, output $\boldsymbol{pk}$.

In our definitions, all keys will *always* be random variables distributed (as key-pairs) according to Σ_{PS}'s $\mathtt{Gen}$.

We begin by defining a combined notion for bilateral signatures capturing both authenticity and unambiguity at once. For this, we define a distinction problem between a real system that correctly generates and verifies signatures, via signing and verification oracles for n (different) senders, and an ideal system that also correctly generates signatures for n (different) senders, but only correctly verifies signatures previously signed by *any* signing oracle.

Definition 7 (UF-UA-Secure Partial Signature). *A partial signature scheme Σ_{PS} is (n, ε)-unforgeable-and-unambiguous (or (n, ε)-UF-UA-secure) if*

$$[\langle \mathbf{S}_{sk_1}, \mathbf{V}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \approx_{\varepsilon} [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{\boldsymbol{pk}}],$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \mathtt{Gen}$ and $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$.

As we formally show in Appendix D.2, it is easy to see that if a partial signature scheme is ε-UF-secure *and* (n, ε')-UA-secure (as defined there), then it is $(n, n \cdot \varepsilon^{\mathbf{C}} + \varepsilon')$-UF-UA-secure, for a specific reduction $\mathbf{C}$.

We next define anonymity of partial signatures. For this, we define a distinction problem between a real system that correctly generates *only* stubs, via (reduced) signing oracles for n (different) senders, and an ideal system that also correctly generates only stubs, but via n copies of (reduced) signing oracles for the *same* (randomly selected) sender.

Definition 8 (IK-Secure Partial Signature). *A partial signature scheme* Σ_{PS} *is* (n, ε)-anonymous *(or* (n, ε)-IK-secure*) if*

$$[\mathbf{S}^{-}_{sk_1}, \ldots, \mathbf{S}^{-}_{sk_n}, \mathbf{K}_{\boldsymbol{pk}}] \approx_{\varepsilon} [\mathbf{S}^{-}_{sk_I}, \ldots, \mathbf{S}^{-}_{sk_I}, \mathbf{K}_{\boldsymbol{pk}}]$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \texttt{Gen}$*,* $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$*, and random variable* $I \xleftarrow{\$} [n]$*.*

Unlike what we did for bilateral signatures (and will later do for ring signatures as well), it is not possible to define a combined security notion for partial signatures capturing both UF-UA-security and IK-security at once. This is because a unified distinction problem would necessarily require a full signing oracle, in order to model unforgeability, thus making it possible to trivially distinguish signatures generated by different senders, that is, making the modeling of anonymity impossible.

4.2 Composable Security of Partial Signatures

As it is made clear by the concrete construction given in [BD09], partial signature schemes inherently involve a special form of commitment. In fact, such straightforward construction from a regular signature scheme and a commitment scheme involves generating a normal signature on the message, and committing to it and the verification key. The resulting commitment bitstring will then be the stub σ (the one ensuring anonymity, but not authenticity), and the opening (or "decommital key") will correspond to the tag τ (the one ensuring authenticity, but not anonymity). More details are found in Appendix E.2.

From this, it becomes immediately apparent that trying to capture security of partial signatures in a composable fashion, would necessarily incur the so-called *simulator commitment problem*. In this specific case, the issue is as follows: Intuitively, in the real world a sender S_i, for $i \in [n]$, generates a full signature (σ, τ) on a message m, and in a first phase sends only (m, σ) to the receiver R, while in a second phase it sends (m, σ, τ), which must satisfy $\texttt{Vrf}_{pk_i}(m, \sigma, \tau) = 1$. But in the ideal world, during the first phase the simulator only receives the message m from $\mathsf{D\text{-}AUT}_{n \to 1}$, and does not know who the sender is (in particular, it does not know the value $i \in [n]$). Even though it emulates all n secret/public keys sk_i, pk_i of the senders, it must output a partial signature σ by producing a full signature (σ, τ) for m using a *different* random secret key sk (this difference in the real and ideal worlds is what exactly captures anonymity of the stub σ). In the second phase, once it obtains the identity i of the sender S_i who sent m, the simulator must be able to output, along with the previously defined stub σ, a valid tag τ that satisfies $\texttt{Vrf}_{pk_i}(m, \sigma, \tau) = 1$. But because upon generation of σ from m, the simulator did not use sk_i, it is infeasible for it to correctly generate such a valid τ.

Recently, a generic workaround to this problem was put forth by Jost and Maurer [JM20], where the use of a new type of relaxation, the so-called *interval-wise relaxation*, allows to make formal statements capturing security notions that in regular composability frameworks would incur in the commitment problem. The interval-wise relaxation builds upon the combination of two other relaxations, the from-relaxation and the until-relaxation. Informally, given a resource $\mathbf{R}$ and two monotone[2] predicates P_1, P_2 (on the history of events happening globally in an experiment involving $\mathbf{R}$), the from-relaxation $\mathbf{R}^{[P_1}$ consists of all resources behaving arbitrarily until P_2 is true and exactly as $\mathbf{R}$ afterwards, whereas the until-relaxation $\mathbf{R}^{P_2]}$ consists of all resources behaving exactly as $\mathbf{R}$ until P_1 is true and arbitrarily afterwards. Hence, intuitively the combined relaxation $\mathbf{R}^{[P_1,P_2]}$ consist of all resources behaving exactly as $\mathbf{R}$ from when P_1 is true and until P_2 is true, and arbitrarily otherwise (technically, it actually corresponds to the transitive closure of taking the from- and until-relaxation in alternating order). Finally, for a function $\varepsilon : \Theta_n \to [0,1]$, the interval-wise relaxation $\mathbf{R}^{[P_1,P_2]:\varepsilon}$ informally corresponds to all resources in $\mathbf{R}^{[P_1,P_2]}$ that are also ε-close to $\mathbf{R}$. Formally, this is defined using the ε-relaxation introduced in Section 2.2 as $\mathbf{R}^{[P_1,P_2]:\varepsilon} \doteq ((\mathbf{R}^{[P_1,P_2]})^{\varepsilon})^{[P_1,P_2]}$ (see [JM20] for more details).

Recall that we want to define composable security of a partial signature scheme Σ_{PS} as the construction of the resource $\mathsf{D\text{-}AUT}_{n\to 1}$ from the resources $\mathsf{1\text{-}AUT}_{n\to 1}$, and $\mathsf{A\text{-}INS}_{n\to 1}$ (instantiated with $\mathcal{X} = (\{\underline{\texttt{cmt}}\} \times \mathcal{M} \times \mathcal{S}) \cup (\{\underline{\texttt{aut}}\} \times \mathcal{M} \times \mathcal{S} \times \mathcal{T})$, referring to Appendix A). In order to make this statement formal, we need to define how a protocol π_{PS}, attached to the resource $[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{A\text{-}INS}_{n\to 1}]$, naturally makes use of Σ_{PS}. First, π_{PS} runs $\texttt{Gen}$ for every sender S_i, for $i \in [n]$, generating key-pairs $(sk_1, pk_1), \dots, (sk_n, pk_n)$. Then it transmits the public keys $pk_1, \dots, pk_n$ to the receiver through $\mathsf{1\text{-}AUT}_{n\to 1}$. After that, for each sender S_i it sets up two look-up tables, modeled here as sets $\mathfrak{H}_i \subseteq \mathbb{N} \times \mathcal{M} \times \mathcal{S} \times \mathcal{T}$ and $\mathfrak{H}'_i \subseteq \mathbb{N} \times \mathcal{M} \times \mathcal{S}$, as well as a handle value $h_i \in \mathbb{N}$, initially set to 0. Then sender S_i might input messages of two different types on its interface:

- $(\underline{\texttt{cmt}}, m)$, for some $m \in \mathcal{M}$: in this case, π_{PS} uses sk_i to generate $(\sigma, \tau) \leftarrow \texttt{Sgn}_{sk_i}(m)$, and inputs $(\underline{\texttt{cmt}}, m, \sigma)$ to the interface S_i of $\mathsf{A\text{-}INS}_{n\to 1}$. Then it sets $h_i \leftarrow h_i + 1$ and $\mathfrak{H}_i \leftarrow \mathfrak{H}_i \cup \{(h_i, m, \sigma, \tau)\}$.
- $(\underline{\texttt{aut}}, h)$, for some $h \in \mathbb{N}$: in this case, π_{PS} first checks whether $(h, m, \sigma, \tau) \in \mathfrak{H}_i$, for some m, σ, τ. If that is the case, then π_{PS} inputs $(\underline{\texttt{aut}}, m, \sigma, \tau)$ to the interface S_i of $\mathsf{A\text{-}INS}_{n\to 1}$.

Once the receiver R inputs $\diamond$ on its interface, π_{PS} also inputs $\diamond$ to the interface R of $\mathsf{A\text{-}INS}_{n\to 1}$, obtaining a set $\mathfrak{O} \subseteq (\mathbb{N} \times \{\underline{\texttt{cmt}}\} \times \mathcal{M} \times \mathcal{S}) \cup (\mathbb{N} \times \{\underline{\texttt{aut}}\} \times \mathcal{M} \times \mathcal{S} \times \mathcal{T})$. Then it sets $\mathfrak{H}' \leftarrow \mathfrak{H}' \cup \{(j, m, \sigma) \mid (j, (\underline{\texttt{cmt}}, m, \sigma)) \in \mathfrak{O}\}$, computes the sets $\mathfrak{O}' \doteq \{(\underline{\texttt{cmt}}, j, m) \mid \exists \sigma \in \mathcal{S} : (j, (\underline{\texttt{cmt}}, m, \sigma)) \in \mathfrak{O}\}$, $\mathfrak{O}'' \doteq \{(\underline{\texttt{aut}}, j', j, i) \mid \exists m \in \mathcal{M}, \sigma \in \mathcal{S}, \tau \in \mathcal{T} : (j', \underline{\texttt{aut}}, m, \sigma, \tau) \in \mathfrak{O}, (j, m, \sigma) \in \mathfrak{H}', \texttt{Vrf}_{pk_i}(m, \sigma, \tau) = 1\}$, and outputs the set $\mathfrak{O}' \cup \mathfrak{O}''$ to R. We call π_{PS} the protocol using Σ_{PS} in the *natural way*.

Intuitively, we model composable security of a partial signature scheme by making a statement for each interval defined by a sequence of inputs at the sender interfaces $\{S_i\}_{i=1}^{n}$ that are of the same type, that is, either all are of the form $(\underline{\texttt{cmt}}, \cdot)$ (*messages*), or all are of the form $(\underline{\texttt{aut}}, \cdot)$ (*handles*). This way, we make

[2] A monotone predicate is a predicate that once becomes true cannot be false anymore.

sure that the individual security statement is within an interval in which the simulator cannot incur the commitment problem. For this we define the following predicates:

- $P_{\mathsf{msg}(j)}$: true if j-th sender input is a *message* m (E would obtain (m, σ));
- $P_{\mathsf{hnd}(j)}$: true if j-th sender input is a *handle* h (E would obtain (m, σ, τ));
- $P_{\mathsf{fst}(j)}$: true at *first* consecutive sender input of *same type as the j-th*;
- $P_{\mathsf{lst}(j)}$: true at *last* consecutive sender input of *same type as the j-th.*

Definition 9. *A partial signature scheme Σ_{PS} is $(n, t, \varepsilon_{\mathsf{m}}, \varepsilon_{\mathsf{h}})$*-composably secure *if for all $\mathcal{C} \subseteq \{S_i\}_{i=1}^n$,*

$$\pi_{\mathsf{PS}}^{\overline{\mathcal{C}}}[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{A\text{-}INS}_{n\to 1}]^{*\mathcal{C}} \subseteq \bigcap\nolimits_{(P_1,P_2,\varepsilon)\in\Omega} (\mathsf{D\text{-}AUT}^{*\mathcal{C}\cup\{E\}}_{n\to 1})^{[P_1,P_2]:\varepsilon},$$

for $\Omega = \{(P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}, \varepsilon_{\mathsf{m}})\}_{j\in[t]:P_{\mathsf{msg}(j)}} \cup \{(P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}, \varepsilon_{\mathsf{h}})\}_{j\in[t]:P_{\mathsf{hnd}(j)}}$*, where $t \in \mathbb{N}$ is an upper-bound on the number of transmitted messages and π_{PS} is the protocol using Σ_{PS} in the natural way.*

Finally, we show that game-based security of partial signatures implies their composable security (we defer the proof to Appendix F.2).

Theorem 2. *There exist reduction systems $\mathbf{C}_{\mathsf{m}}$ and $\mathbf{C}_{\mathsf{h}}$ such that, if a partial signature scheme Σ_{PS} is $(n, \varepsilon_{\mathsf{m}})$-*IK*-secure and $(n, \varepsilon_{\mathsf{h}})$-*UF-UA*-secure, then it is $(n, t, \varepsilon_{\mathsf{m}}^{\mathbf{C}_{\mathsf{m}}}, \varepsilon_{\mathsf{h}}^{\mathbf{C}_{\mathsf{h}}})$-composably secure, for any $t \in \mathbb{N}$.*

Remark 1. It is natural to ask whether regular signatures would also achieve the notion of Definition 9. This would correspond to asking whether a partial signature scheme with empty strings as stubs would still satisfy Theorem 2. The short answer is no, because it is easy to see that such a scheme does not necessarily achieve unambiguity. Nevertheless, we point out that in principle it should be possible to construct unambiguous regular signature schemes, but still we chose to use partial signatures instead because they offer more: If the adversary also publishes its public-key, then non-empty stubs and unambiguity ensure that it cannot falsely claim any message of the honest senders. This would follow trivially by appropriately extending our definitions, but it would not if a regular signature scheme was used instead. We leave the problem of formalizing this variant open for future work.

5 Achieving Receiver-Side Anonymous Authenticity

One of the first alternative signature schemes providing some form of anonymity were *group signatures*, introduced by Chaum and Van Heyst [CvH91]. The main idea is that members of a group share a public verification key, which can be used to verify a message-signature pair generated by any of the group members using their own (different) secret keys. Anonymity is enforced by ensuring that the verification process does not reveal any partial information about the secret key used to generate the signature, hence effectively allowing a member to anonymously sign a message on behalf of the group. Technically, this is achieved by assigning the role of group manager to a selected member, which is responsible for generating all members' secret keys as well as the group's public verification key. Therefore, the group manager also has the ability to reveal the original signer.

This drawback of group signatures was later circumvented by Rivest, Shamir, and Tauman [RST01], who introduced *ring signature*. In this new scheme, a signature is generated by using not only the sender secret key, but also all the public keys of the group's members, called a ring in this context. Therefore, a signature must be transmitted along with the list of all public keys used, and anonymity is again enforced by requiring that the verification process does not reveal any partial information about the secret key used to generate the signature. Another advantage of ring signatures, compared to group signatures, is that the ring can be dynamically chosen by the sender, and does not require any cooperation.

The syntax of a ring signature scheme, for a fixed ring size of $n \in \mathbb{N}$, extends that of a regular DSS as follows: each sender generates its key-pair (sk_i, pk_i), for $i \in [n]$, but in order to generate a signature σ on a message m, in addition to sk_i, the list $\boldsymbol{pk} \doteq (pk_1, \dots, pk_n)$ of all other senders public keys is needed. Moreover, also the index i itself is required by the signing function, in order to link the given secret key to the public key of the sender. Then, the receiver can verify that σ is a valid signature for m by using $\boldsymbol{pk}$, and be assured that the message was authentically transmitted by one of the known senders, and no external adversary.

Definition 10 (Ring Signature Scheme). *A* ring signature scheme (RSS) $\Sigma_{\mathsf{RS}} \doteq (\mathtt{Gen}, \mathtt{Sgn}, \mathtt{Vrf})$ *for $n \geq 2$ users over message-space $\mathcal{M}$ and signature-space $\mathcal{S}$ (with $\bot \notin \mathcal{M} \cup \mathcal{S}$), is such that*

- $\mathtt{Gen}$ *is a distribution over the key-space $\mathcal{SK} \times \mathcal{PK}$;*
- $\mathtt{Sgn} : [n] \times \mathcal{SK} \times \mathcal{PK}^n \times \mathcal{M} \to \mathcal{S}$ *is a probabilistic function;*
- $\mathtt{Vrf} : \mathcal{PK}^n \times \mathcal{M} \times \mathcal{S} \to \{0,1\}$ *is a deterministic function.*

We require the above to be efficiently samplable/computable. For index $i \in [n]$ and keys $sk \in \mathcal{SK}$, $\boldsymbol{pk} \doteq (pk_1, \dots, pk_n) \in \mathcal{PK}^n$, for any $n \in \mathbb{N}$, we use the shorthand notation $\mathtt{Sgn}_{i,sk,\boldsymbol{pk}}(\cdot)$ for $\mathtt{Sgn}(i, sk, \boldsymbol{pk}, \cdot)$ and $\mathtt{Vrf}_{\boldsymbol{pk}}(\cdot,\cdot)$ for $\mathtt{Vrf}(\boldsymbol{pk}, \cdot, \cdot)$. Moreover, we assume correctness *of Σ_{RS}, that is, for all $n \geq 2$, all $i \in [n]$, all possible lists of n key-pairs $(sk_1, pk_1), \dots, (sk_n, pk_n)$ distributed according to* $\mathtt{Gen}$*, with $\boldsymbol{pk} \doteq (pk_1, \dots, pk_n)$, all messages $m \in \mathcal{M}$, and all signatures $\sigma \in \mathcal{S}$,* $\mathtt{Vrf}_{\boldsymbol{pk}}(m, \sigma) = \mathbb{1}\left\{\sigma \in \bigcup_{i=1}^{n} \operatorname{supp}\left(\mathtt{Sgn}_{i,sk_i,\boldsymbol{pk}}(m)\right)\right\}$.

5.1 Game-Based Security of Ring Signatures

When ring signatures were introduced in [RST01], no formal game-based security definitions were given, this was only done later in [BKM06]. There, a stronger model than the one considered here was introduced, namely one where the adversary can generate and publish its own public key, which, as discussed in

Section 2.3, would require a certificate authority. Therefore, here we use adapted versions of the weaker security notions of *unforgeability against fixed-ring attacks* and *basic anonymity* from [BKM06]. In Appendices D.3 and E.3 we relate those notions from the literature to the new combined definition we introduce next.

In order to define the security of a fixed scheme Σ_{RS}, we define the following systems (where the dependency on Σ_{RS} is implicit), parameterized by index $i \in [n]$ and keys $sk \in \mathcal{SK}$, $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n) \in \mathcal{PK}^n$, for any $n \in \mathbb{N}$.

- $\langle \mathbf{S}_{i,sk,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle$:
  - On input $m \in \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $\sigma \leftarrow \mathtt{Sgn}_{i,sk,\boldsymbol{pk}}(m)$.
  - On input $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, return m if $\mathtt{Vrf}_{\boldsymbol{pk}}(m, \sigma) = 1$ and $\perp$ otherwise.
- $\langle \mathbf{S}_{i,sk,\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle$: Set $\mathcal{Q} \subseteq \mathcal{M} \times \mathcal{S}$ to $\varnothing$, and then:
  - On input $m \in \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $\sigma \leftarrow \mathtt{Sgn}_{i,sk,\boldsymbol{pk}}(m)$, and set $\mathcal{Q}$ to $\mathcal{Q} \cup \{(m, \sigma)\}$.
  - On input $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, return m if $(m, \sigma) \in \mathcal{Q}$ and $\perp$ otherwise.
- $\mathbf{K}_{\boldsymbol{pk}}$: On input $\diamond$, output $\boldsymbol{pk}$.

In our definitions, all keys will *always* be random variables distributed (as key-pairs) according to Σ_{RS}'s $\mathtt{Gen}$.

We define a combined notion for ring signatures capturing both authenticity and anonymity at once. For this, we define a distinction problem between a real system that correctly generates and verifies signatures, via signing and verification oracles for n (different) senders, and an ideal system that also correctly generates signatures and only correctly verifies signatures previously signed, but via n copies of signing and verification oracles for the *same* (randomly selected) sender.

Definition 11 (UF-IK-Secure Ring Signature). *A ring signature scheme Σ_{RS} is (n, ε)-unforgeable-and-anonymous (or (n, ε)-UF-IK-secure) if*

$$[\langle \mathbf{S}_{1,sk_1,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \ldots, \langle \mathbf{S}_{n,sk_n,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \mathbf{K}_{\boldsymbol{pk}}]$$
$$\approx_{\varepsilon}$$
$$[\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}],$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \mathtt{Gen}$, $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$, and random variable $I \overset{\$}{\leftarrow} [n]$.

As we formally show in Appendix D.3, it is easy to see that if a ring signature scheme is (n, ε)-UF-secure *and* (n, ε')-IK-secure (as defined there), then it is $(n, \varepsilon^{\mathbf{C}} + \varepsilon')$-UF-IK-secure, for a specific reduction $\mathbf{C}$.

5.2 Composable Security of Ring Signatures

We continue our study of the semantics of ring signatures by defining their composable security in the constructive cryptography framework. Composable security notions for ring signatures have been previously studied in [YO07] within the universal composability (UC) framework. There, an ideal functionality was introduced, and it was shown to be securely realized by a protocol employing ring signatures. Unlike with our approach, such functionality was completely tailored to the ring signature scheme used by the protocol, that is, it exported operations such as signing and verifying, it did not model a communication channel between senders and receiver. Here we define an ideal resource, independent of any cryptographic scheme, and show that (among other possible ones), a protocol employing ring signatures indeed realizes such a resource.

Recall that we want to define composable security of a ring signature scheme Σ_{RS} as the construction of the resource $\mathsf{RA\text{-}AUT}_{n\rightarrow 1}$ from the resources $\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}$ and $\mathsf{A\text{-}INS}_{n\rightarrow 1}$ (instantiated with $\mathcal{X} = \mathcal{M} \times \mathcal{S}$, referring to Appendix A). In order to make this statement formal, we need to define how a protocol π_{RS}, attached to the resource $[\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}, \mathsf{A\text{-}INS}_{n\rightarrow 1}]$, naturally makes use of Σ_{RS}. First, π_{RS} runs $\texttt{Gen}$ for every sender S_i, for $i \in [n]$, generating key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n)$. Then it transmits the public keys $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$ to the receiver and all senders through $\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}$. After that, once a sender S_i inputs a message m on its interface, π_{RS} uses sk_i and $\boldsymbol{pk}$ to generate $\sigma \leftarrow \texttt{Sgn}_{i,sk_i,\boldsymbol{pk}}(m)$, and inputs (m, σ) to the interface S_i of $\mathsf{A\text{-}INS}_{n\rightarrow 1}$. Once the receiver R inputs $\diamond$ on its interface, π_{RS} also inputs $\diamond$ to the interface R of $\mathsf{A\text{-}INS}_{n\rightarrow 1}$, obtaining a set $\mathfrak{D} \subseteq \mathbb{N} \times \mathcal{M} \times \mathcal{S}$, and outputs the set $\{(j, m) \mid \exists\, (j, m, \sigma) \in \mathfrak{D} : \texttt{Vrf}_{\boldsymbol{pk}}(m, \sigma) = 1\}$ to R. We call π_{RS} the protocol using Σ_{RS} in the *natural way*.

Definition 12. *A ring signature scheme* Σ_{RS} *is* (n, ε)-composably secure *if*

$$[\mathsf{1\text{-}AUT}_{n\circlearrowleft 1}, \mathsf{A\text{-}INS}_{n\rightarrow 1}] \xmapsto{\pi_{\mathsf{RS}}, \varepsilon} \mathsf{RA\text{-}AUT}_{n\rightarrow 1},$$

where π_{RS} *is the protocol using* Σ_{RS} *in the natural way.*

Finally, we show that game-based security of ring signatures implies their composable security (we defer the proof to Appendix F.3).

Theorem 3. *There exists a reduction system* $\mathbf{C}$ *such that, if a ring signature scheme* Σ_{RS} *is* (n, ε)-$\mathsf{UF\text{-}IK}$*-secure, then it is* $(n, \varepsilon^{\mathbf{C}})$*-composably secure.*

6 Concluding Remarks and Future Work

This work focused on filling a gap in the composable treatment of anonymity preservation in the public-key setting. Being of definitional nature, it was centered around providing clear composable semantics of existing schemes, as well as showing how existing and new game-based security notions for such schemes imply composable statements. This is very desirable in order to understand how such schemes should be used in practice.

Still, since the scope of this work was very ample, we see it as merely paving the way. For example, additional alternative solutions circumventing our impossibility result, employing different schemes, might be interesting to analyze. Moreover, all of our results hold under static corruptions, therefore a natural extension would be to consider a stronger security model capturing adaptive corruptions. This would allow to rely on stronger game-based notions from the literature for partial signatures and ring signatures.

References

ABN10. Michel Abdalla, Mihir Bellare, and Gregory Neven. Robust encryption. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 480–497, Heidelberg, 2010. Springer. doi:10.1007/978-3-642-11799-2_28.

AHM$^+$15. Joël Alwen, Martin Hirt, Ueli Maurer, Arpita Patra, and Pavel Raykov. Anonymous authentication with shared secrets. In Diego F. Aranha and Alfred Menezes, editors, *LATINCRYPT 2014*, volume 8895 of *LNCS*, pages 219–236, Cham, 2015. Springer. doi:10.1007/978-3-319-16295-9_12.

BBDP01. Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In Colin Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 566–582, Heidelberg, 2001. Springer. doi:10.1007/3-540-45682-1_33.

BD09. Mihir Bellare and Shanshan Duan. Partial signatures and their applications. Cryptology ePrint Archive, Report 2009/336, 2009. https://eprint.iacr.org/2009/336.

BKM06. Adam Bender, Jonathan Katz, and Ruggero Morselli. Ring signatures: Stronger definitions, and constructions without random oracles. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 60–79, Heidelberg, 2006. Springer. doi:10.1007/11681878_4.

BM20. Fabio Banfi and Ueli Maurer. Anonymous symmetric-key communication. In Clemente Galdi and Vladimir Kolesnikov, editors, *SCN 2020*, volume 12238 of *LNCS*, pages 471–491, Cham, 2020. Springer. doi:10.1007/978-3-030-57990-6_23.

CvH91. David Chaum and Eugène van Heyst. Group signatures. In Donald W. Davies, editor, *EUROCRYPT 1991*, volume 547 of *LNCS*, pages 257–265, Heidelberg, 1991. Springer. doi:10.1007/3-540-46416-6_22.

Fis07. Marc Fischlin. Anonymous signatures made easy. In Tatsuaki Okamoto and Xiaoyun Wang, editors, *PKC 2007*, volume 4450 of *LNCS*, pages 31–42, Heidelberg, 2007. Springer. doi:10.1007/978-3-540-71677-8_3.

JM20. Daniel Jost and Ueli Maurer. Overcoming impossibility results in composable security using interval-wise guarantees. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020*, volume 12170 of *LNCS*, pages 33–62, Cham, 2020. Springer. doi:10.1007/978-3-030-56784-2_2.

JSI96. Markus Jakobsson, Kazue Sako, and Russell Impagliazzo. Designated verifier proofs and their applications. In Ueli Maurer, editor, *EUROCRYPT 1996*, volume 1070 of *LNCS*, pages 143–154, Heidelberg, 1996. Springer. doi:10.1007/3-540-68339-9_13.

KMO$^+$13. Markulf Kohlweiss, Ueli Maurer, Cristina Onete, Björn Tackmann, and Daniele Venturi. Anonymity-preserving public-key encryption: A constructive approach. In Emiliano De Cristofaro and Matthew Wright, editors, *PETS 2013*, volume 7981 of *LNCS*, pages 19–39, Heidelberg, 2013. Springer. doi:10.1007/978-3-642-39077-7_2.

LV05. Fabien Laguillaumie and Damien Vergnaud. Designated verifier signatures: Anonymity and efficient construction from any bilinear map. In Carlo Blundo and Stelvio Cimato, editors, *SCN 2004*, volume 3352 of *LNCS*, pages 105–119, Heidelberg, 2005. Springer. doi:10.1007/978-3-540-30598-9_8.

Mau02. Ueli Maurer. Indistinguishability of random systems. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 110–132, Heidelberg, 2002. Springer. doi:10.1007/3-540-46035-7_8.

Mau12. Ueli Maurer. Constructive cryptography – a new paradigm for security definitions and proofs. In Sebastian Mödersheim and Catuscia Palamidessi, editors, *TOSCA 2011*, volume 6993 of *LNCS*, pages 33–56, Heidelberg, 2012. Springer. doi:10.1007/978-3-642-27375-9_3.

MPR07. Ueli Maurer, Krzysztof Pietrzak, and Renato Renner. Indistinguishability amplification. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 130–149, Heidelberg, 2007. Springer. doi:10.1007/978-3-540-74143-5_8.

MPR21. Ueli Maurer, Christopher Portmann, and Guilherme Rito. Giving an adversary guarantees (or: How to model designated verifier signatures in a composable framework), 2021. doi:10.1007/978-3-030-92078-4_7.

MR11. Ueli Maurer and Renato Renner. Abstract cryptography. In *ICS 2011*, pages 1–21. Tsinghua University Press, 2011.

MR16. Ueli Maurer and Renato Renner. From indifferentiability to constructive cryptography (and back). In Martin Hirt and Adam Smith, editors, *TCC 2016*, volume 9985 of *LNCS*, pages 3–24, Heidelberg, 2016. Springer. doi:10.1007/978-3-662-53641-4_1.

Ros18. Mike Rosulek. The joy of cryptography. Oregon State University EOR, 2018. http://web.engr.oregonstate.edu/~rosulekm/crypto/.

RST01. Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. In Colin Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 552–565, Heidelberg, 2001. Springer. doi:10.1007/3-540-45682-1_32.

SBWP03. Ron Steinfeld, Laurence Bull, Huaxiong Wang, and Josef Pieprzyk. Universal designated-verifier signatures. In Chi-Sung Laih, editor, *ASIACRYPT 2003*, volume 2894 of *LNCS*, pages 523–542, Heidelberg, 2003. Springer. doi:10.1007/978-3-540-40061-5_33.

SWP04. Ron Steinfeld, Huaxiong Wang, and Josef Pieprzyk. Efficient extension of standard Schnorr/RSA signatures into universal designated-verifier signatures. In Feng Bao, Robert Deng, and Jianying Zhou, editors, *PKC 2004*, volume 2947 of *LNCS*, pages 86–100, Heidelberg, 2004. Springer. doi:10.1007/978-3-540-24632-9_7.

SY09. Vishal Saraswat and Aaram Yun. Anonymous signatures revisited. In Josef Pieprzyk and Fangguo Zhang, editors, *ProvSec 2009*, volume 5848 of *LNCS*, pages 140–153, Heidelberg, 2009. Springer. doi:10.1007/978-3-642-04642-1_13.

YO07. Kazuki Yoneyama and Kazuo Ohta. Ring signatures: Universally composable definitions and constructions. *IPSJ Digital Courier*, 3:571–584, 2007. doi:10.2197/ipsjdc.3.571.

YWDW06. Guomin Yang, Duncan S. Wong, Xiaotie Deng, and Huaxiong Wang. Anonymous signature schemes. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin, editors, *PKC 2006*, volume 3958 of *LNCS*, pages 347–363, Heidelberg, 2006. Springer. doi:10.1007/11745853_23.

ZI09. Rui Zhang and Hideki Imai. Strong anonymous signatures. In Moti Yung, Peng Liu, and Dongdai Lin, editors, *Inscrypt 2008*, volume 5487 of *LNCS*, pages 60–71, Heidelberg, 2009. Springer. doi:10.1007/978-3-642-01440-6_7.

A Formal Description of Resources

```
1-AUT$_{n\to 1}$
$k_1, \dots, k_n \in \mathcal{K} \cup \{\bot\}$
Initialize:
    $k_1, \dots, k_n \leftarrow \bot$
Interface $S_i(k \in \mathcal{K})$:
    $k_i \leftarrow k$
Interface $E(\diamond)$:
    $\mathfrak{O} \leftarrow \{(i, k_i) \,|\, i \in [n]\}$
    return $\mathfrak{O}$
Interface $R(\diamond)$:
    $\mathfrak{O} \leftarrow \{(i, k_i) \,|\, i \in [n]\}$
    return $\mathfrak{O}$
```

```
1-AUT$_{n\leftarrow 1}$
$k \in \mathcal{K} \cup \{\bot\}$
Initialize:
    $k \leftarrow \bot$
Interface $S_i(\diamond)$:
    return $k$
Interface $E(\diamond)$:
    return $k$
Interface $R(\kappa \in \mathcal{K})$:
    $k \leftarrow \kappa$
```

```
1-AUT$_{n\circlearrowleft 1}$
$k_1, \dots, k_n \in \mathcal{K} \cup \{\bot\}$
Initialize:
    $k_1, \dots, k_n \leftarrow \bot$
Interface $S_i(k \in \mathcal{K})$:
    $k_i \leftarrow k$
Interface $S_i(\diamond)$:
    $\mathfrak{O} \leftarrow \{(i, k_i) \,|\, i \in [n]\}$
    return $\mathfrak{O}$
Interface $E(\diamond)$:
    $\mathfrak{O} \leftarrow \{(i, k_i) \,|\, i \in [n]\}$
    return $\mathfrak{O}$
Interface $R(\diamond)$:
    $\mathfrak{O} \leftarrow \{(i, k_i) \,|\, i \in [n]\}$
    return $\mathfrak{O}$
```

```
A-INS$_{n\to 1}$
$\mathfrak{S}, \mathfrak{R} \subseteq \mathbb{N} \times \mathcal{X},\ c_S, c_R, t_S, t_R \in \mathbb{N}$
Initialize:
    $\mathfrak{S}, \mathfrak{R} \leftarrow \varnothing,\ c_S, c_R \leftarrow 1,\ t_S, t_R \leftarrow 0$
Interface $S_i(x \in \mathcal{X})$:
    $t_S \leftarrow t_S + 1,\ \mathfrak{S} \leftarrow \mathfrak{S} \cup \{(t_S, x)\}$
Interface $E(\diamond)$:
    $\mathfrak{O} \leftarrow \{(j, x) \in \mathfrak{S} \,|\, c_S \leq j \leq t_S\},\ c_S \leftarrow t_S + 1$
    return $\mathfrak{O}$
Interface $E(x \in \mathcal{X})$:
    $t_R \leftarrow t_R + 1,\ \mathfrak{R} \leftarrow \mathfrak{R} \cup \{(t_R, x)\}$
Interface $R(\diamond)$:
    $\mathfrak{O} \leftarrow \{(j, x) \in \mathfrak{R} \,|\, c_R \leq j \leq t_R\},\ c_R \leftarrow t_R + 1$
    return $\mathfrak{O}$
```

```
A-AUT$_{n\to 1}$
$\mathfrak{S}, \mathfrak{R} \subseteq (\mathbb{N} \times \mathcal{M} \times \mathbb{N}) \cup (\mathbb{N} \times \{\bot\}^2),\ c_S, c_R, t_S, t_R \in \mathbb{N}$
Initialize:
    $\mathfrak{S}, \mathfrak{R} \leftarrow \varnothing,\ c_S, c_R \leftarrow 1,\ t_S, t_R \leftarrow 0$
Interface $S_i(m \in \mathcal{M})$:
    $t_S \leftarrow t_S + 1,\ \mathfrak{S} \leftarrow \mathfrak{S} \cup \{(t_S, m, i)\}$
Interface $E(\diamond)$:
    $\mathfrak{O} \leftarrow \{(j, m) \in \mathbb{N} \times \mathcal{M} \,|\, \exists i \in [n] : (j, m, i) \in \mathfrak{S},\ c_S \leq j \leq t_S\},\ c_S \leftarrow t_S + 1$
    return $\mathfrak{O}$
```

```
Interface $E(j \in \mathbb{N})$:
    if $\exists\, m \in \mathcal{M}, i \in [n] : (j, m, i) \in \mathfrak{S}$ then
        $t_R \leftarrow t_R + 1,\ \mathfrak{R} \leftarrow \mathfrak{R} \cup \{(t_R, m, i)\}$
Interface $R(\diamond)$:
    $\mathfrak{O} \leftarrow \{(j, m, i) \in \mathfrak{R} \,|\, c_R \leq j \leq t_R\},\ c_R \leftarrow t_R + 1$
    return $\mathfrak{O}$
```

D-AUT$_{n\to 1}$

```
$\mathfrak{S} \subseteq (\{\underline{\texttt{cmt}}\} \times \mathbb{N} \times \mathcal{M} \times [n] \times \mathbb{N}) \cup (\{\underline{\texttt{aut}}\} \times \mathbb{N}^2 \times \mathcal{M} \times [n])$,
$\mathfrak{R} \subseteq (\{\underline{\texttt{cmt}}\} \times \mathbb{N} \times \mathcal{M}) \cup (\{\underline{\texttt{aut}}\} \times \mathbb{N}^2 \times [n])$,
$\mathfrak{L} \subseteq \mathbb{N}^2,\ c_S, c_R, t_S, t_R, h_1, \ldots, h_n \in \mathbb{N}$
Initialize:
    $\mathfrak{S}, \mathfrak{R}, \mathfrak{L} \leftarrow \varnothing,\ c_S, c_R \leftarrow 1,\ t_S, t_R, h_1, \ldots, h_n \leftarrow 0$
Interface $S_i(\underline{\texttt{cmt}}, m \in \mathcal{M})$:
    $t_S \leftarrow t_S + 1,\ h_i \leftarrow h_i + 1,\ \mathfrak{S} \leftarrow \mathfrak{S} \cup \{(\underline{\texttt{cmt}}, t_S, m, i, h_i)\}$
    return $h_i$
Interface $S_i(\underline{\texttt{aut}}, h \in \mathbb{N})$:
    if $\exists\, j \in \mathbb{N}, m \in \mathcal{M} : (\underline{\texttt{cmt}}, j, m, i, h) \in \mathfrak{S}$ then
        $t_S \leftarrow t_S + 1,\ \mathfrak{S} \leftarrow \mathfrak{S} \cup \{(\underline{\texttt{aut}}, t_S, j, m, i)\}$
Interface $E(\diamond)$:
    $\mathfrak{O} \leftarrow \{(\underline{\texttt{cmt}}, j \in \mathbb{N}, m \in \mathcal{M}) \,|\, \exists\, i \in [n], h \in \mathbb{N} : (\underline{\texttt{cmt}}, j, m, i, h) \in \mathfrak{S},$
            $c_S \leq j \leq t_S\} \cup \{(\underline{\texttt{aut}}, j, j', m, i) \in \mathfrak{S} \,|\, c_S \leq j \leq t_S\},\ c_S \leftarrow t_S + 1$
    return $\mathfrak{O}$
Interface $E(j \in \mathbb{N})$:
    if $\exists\, m \in \mathcal{M}, i \in [n], h \in \mathbb{N} : (\underline{\texttt{cmt}}, j, m, i, h) \in \mathfrak{S}$ then
        $t_R \leftarrow t_R + 1,\ \mathfrak{R} \leftarrow \mathfrak{R} \cup \{(\underline{\texttt{cmt}}, t_R, m)\},\ \mathfrak{L} \leftarrow \mathfrak{L} \cup \{(j, t_R)\}$
    else if $\exists\, j' \in \mathbb{N}, m \in \mathcal{M}, i \in [n] : (\underline{\texttt{aut}}, j, j', m, i) \in \mathfrak{S}$ then
        if $\exists\, j'' \in \mathbb{N} : (j', j'') \in \mathfrak{L}$ then
            $t_R \leftarrow t_R + 1,\ \mathfrak{R} \leftarrow \mathfrak{R} \cup \{(\underline{\texttt{aut}}, t_R, j'', i)\}$
Interface $E(\underline{\texttt{cmt}}, m \in \mathcal{M})$:
    $t_R \leftarrow t_R + 1,\ \mathfrak{R} \leftarrow \mathfrak{R} \cup \{(\underline{\texttt{cmt}}, t_R, m)\}$
Interface $R(\diamond)$:
    $\mathfrak{O} \leftarrow \{(\underline{\texttt{cmt}}, j, m), (\underline{\texttt{aut}}, j, j', i) \in \mathfrak{R} \,|\, c_R \leq j \leq t_R\},\ c_R \leftarrow t_R + 1$
    return $\mathfrak{O}$
```

RA-AUT$_{n\to 1}$

```
$\mathfrak{S}, \mathfrak{R} \subseteq (\mathbb{N} \times \mathcal{M}) \cup (\mathbb{N} \times \{\bot\}),\ c_S, c_R, t_S, t_R \in \mathbb{N}$
Initialize:
    $\mathfrak{S}, \mathfrak{R} \leftarrow \varnothing,\ c_S, c_R \leftarrow 1,\ t_S, t_R \leftarrow 0$
Interface $S_i(m \in \mathcal{M})$:
    $t_S \leftarrow t_S + 1,\ \mathfrak{S} \leftarrow \mathfrak{S} \cup \{(t_S, m)\}$
Interface $E(\diamond)$:
    $\mathfrak{O} \leftarrow \{(j,m) \in \mathfrak{S} \,|\, c_S \leq j \leq t_S\},\ c_S \leftarrow t_S + 1$
    **return** $\mathfrak{O}$
**Interface** $E(j \in \mathbb{N})$:
    **if** $\exists\, m \in \mathcal{M} : (j,m) \in \mathfrak{S}$ **then**
        $t_R \leftarrow t_R + 1,\ \mathfrak{R} \leftarrow \mathfrak{R} \cup \{(t_R, m)\}$
**Interface** $R(\diamond)$:
    $\mathfrak{O} \leftarrow \{(j,m) \in \mathfrak{S} \,|\, c_R \leq j \leq t_R\},\ c_R \leftarrow t_R + 1$
    **return** $\mathfrak{O}$
```

B Impossibility of Anonymity Preservation from DSS

In this section we briefly formalize the simple intuition that regular digital signature schemes (DSS) do not preserve anonymity. We do so in a more generic and composable way: What we prove is that actually no protocol can enhance an insecure channel to an authentic one while preserving its anonymity by only having public one-time authentic information flowing from the receivers to the sender. Clearly, using DSS in the usual way is just one of the possible such protocols.

Proposition 1. *For any protocol π, any corruption set $\mathcal{C}$, and any $\varepsilon < 1 - \frac{1}{n}$,*

$$\pi^{\overline{\mathcal{C}}}[\textsf{1-AUT}_{n\to 1}, \textsf{A-INS}_{n\to 1}]^{*\mathcal{C}} \not\sqsubseteq ((\textsf{A-AUT}_{n\to 1})^{*\mathcal{C}\cup\{E\}})^{\varepsilon}.$$

Proof. Let π be any n-protocol and $\textsf{sim}$ any simulator. We prove the statement by showing that there is a distinguisher $\mathbf{D}$ such that

$$\Delta^{\mathbf{D}}(\pi[\textsf{1-AUT}_{n\to 1}, \textsf{A-INS}_{n\to 1}], \textsf{sim}^E\, \textsf{A-AUT}_{n\to 1}) \geq 1 - \frac{1}{n},$$

where we choose the corruption set $\mathcal{C} = \varnothing$, which is sufficient to prove the claim. $\mathbf{D}$ works as follows. First, it chooses a random message $m \xleftarrow{\$} \mathcal{M}$ and a random index $i \xleftarrow{\$} [n]$, and inputs m at interface S_i. Then it inputs $\diamond$ at interface E of (possibly emulated) $\textsf{A-INS}_{n\to 1}$, and obtains[3] $(0, m, \sigma)$. It subsequently inputs i at interface E of (possibly emulated) $\textsf{1-AUT}_{n\to 1}$, obtains (possibly emulated) public one-time authentic value pk_i, and then emulates the (fixed and publicly known) protocol π on input pk_i and (m, σ) at interface R. Finally, $\mathbf{D}$ outputs 0 if and only if it obtains (m, i) from its emulation. We now analyze two cases. First, assume that $\mathbf{D}$ is interacting with the real-world resource $\pi[\textsf{1-AUT}_{n\to 1}, \textsf{A-INS}_{n\to 1}]$. Then by the correctness of π, $\mathbf{D}$ will obtain (m, i) with probability 1 from its emulation. On the other hand, if $\mathbf{D}$ is interacting with the ideal-resource $\textsf{sim}^E\, \textsf{A-AUT}_{n\to 1}$ instead, then $\mathbf{D}$ will obtain (m, i) with probability at most $\frac{1}{n}$ from its emulation. This is because $\textsf{sim}$ has no better choice than to actually emulate π as well, and choose at random one of the n public one-time authentic values from emulated $\textsf{A-AUT}_{n\to 1}$ to generate σ (since it does not obtain the index of the sender from $\textsf{A-AUT}_{n\to 1}$). Therefore, $\mathbf{D}$'s advantage is at least $1 - \frac{1}{n}$. □

[3] Note that we are assuming (w.l.o.g.) that π always transmits m.

C Composability of Anonymous Constructions

First of all, note that the construction notion from Definition 1 directly implies composability via the transitivity of the subset relation.

Lemma 1. *For any specifications $\mathcal{R}, \mathcal{S}, \mathcal{T} \subseteq \Phi$ and constructors $\gamma, \gamma' : \Phi \to \Phi$,*

$$\mathcal{R} \xrightarrow{\gamma} \mathcal{S} \;\wedge\; \mathcal{S} \xrightarrow{\gamma'} \mathcal{T} \quad \Longrightarrow \quad \mathcal{R} \xrightarrow{\gamma' \circ \gamma} \mathcal{T}.$$

Proof. $\gamma' \circ \gamma(\mathcal{R}) \subseteq \gamma'(\mathcal{S}) \subseteq \mathcal{T}$. □

We are interested in relaxations that are *compatible* with constructors, that is, for a constructor γ we consider a relaxation ρ to be compatible if there exists another relaxation ρ^γ depending on γ, where the superscript makes dependency on γ explicit, such that $\gamma(\mathcal{S}^\rho) \subseteq \gamma(\mathcal{S})^{\rho^\gamma}$, for any specification $\mathcal{S}$. In this case we say that ρ *is compatible with* γ *via* ρ^γ. This property can be interpreted as a special kind of commutativity, since it is equivalent to $\gamma \circ \rho(\mathcal{S}) \subseteq \rho^\gamma \circ \gamma(\mathcal{S})$. All relaxation considered in this work satisfy this property, and we will later show exactly how (that is, for a specific relaxation ρ and arbitrary constructor γ, we will explicitly construct ρ^γ). We now show how this property is useful to show composability in general.

Lemma 2. *Let $\rho : \Phi \to 2^\Phi$ be a relaxation compatible with constructor $\gamma : \Phi \to \Phi$ via relaxation $\rho^\gamma : \Phi \to 2^\Phi$, that is, $\gamma(\mathcal{S}^\rho) \subseteq \gamma(\mathcal{S})^{\rho^\gamma}$, for any specification $\mathcal{S} \subseteq \Phi$. Then we have*

$$\mathcal{R} \xrightarrow{\gamma} \mathcal{S} \quad \Longrightarrow \quad \mathcal{R}^\rho \xrightarrow{\gamma} \mathcal{S}^{\rho^\gamma}.$$

Proof. $\gamma \circ \rho(\mathcal{R}) \subseteq \rho^\gamma \circ \gamma(\mathcal{R}) \subseteq \rho^\gamma(\mathcal{S})$. □

This fact allows us to compose two construction statements where the assumed specification of the second statement appears relaxed as the constructed specification of the first statement. Concretely, using Lemma 1 and Lemma 2, we have

$$\mathcal{R} \xrightarrow{\gamma} \mathcal{S}^\rho \;\wedge\; \mathcal{S} \xrightarrow{\gamma'} \mathcal{T} \quad \Longrightarrow \quad \mathcal{R} \xrightarrow{\gamma' \circ \gamma} \mathcal{T}^{\rho^{\gamma'}}. \tag{2}$$

In our specific setting, since constructors are instantiated by n-protocols, the compatibility condition $\gamma(\mathcal{S}^\rho) \subseteq (\gamma(\mathcal{S}))^{\rho^\gamma}$ from Lemma 2 translates into $\pi\mathcal{S}^\rho \subseteq (\pi\mathcal{S})^{\rho^\pi}$. Moreover, it is easily verified that:

- For any $\mathcal{C} \subseteq \{S_i\}_{i=1}^n$, the $*_\mathcal{C}$-relaxation is compatible with $\pi^{\overline{\mathcal{C}}}$, for any $\pi \in \Sigma^{n+1}$, via $*_\mathcal{C}$: For any specification $\mathcal{S} \subseteq \Phi_n$, we have

$$\begin{aligned}
\pi^{\overline{\mathcal{C}}}\mathcal{S}^{*_\mathcal{C}} &= \pi^{\overline{\mathcal{C}}}\{\alpha^\mathcal{C}\mathbf{S} \,|\, \mathbf{S} \in \mathcal{S}, \alpha \in \Sigma^{n+1}\} \\
&= \{\pi^{\overline{\mathcal{C}}}\alpha^\mathcal{C}\mathbf{S} \,|\, \mathbf{S} \in \mathcal{S}, \alpha \in \Sigma^{n+1}\} \\
&= \{\alpha^\mathcal{C}(\pi^{\overline{\mathcal{C}}}\mathbf{S}) \,|\, \mathbf{S} \in \mathcal{S}, \alpha \in \Sigma^{n+1}\} \\
&= (\pi^{\overline{\mathcal{C}}}\mathcal{S})^{*_\mathcal{C}},
\end{aligned}$$

  where the third step follows from the fact that $\overline{\mathcal{C}} \cap \mathcal{C} = \varnothing$.

- For any $\varepsilon : \Theta_n \to [0,1]$, if we define $\varepsilon^{\pi}(\mathbf{D}) \doteq \varepsilon(\mathbf{D}\pi)$, for any distinguisher $\mathbf{D} \in \Theta_n$, then the ε-relaxation is compatible with any $\pi \in \Sigma^{n+1}$ via ε^{π}: For any specification $\mathcal{S} \subseteq \Phi_n$, we have

$$\begin{aligned}
\pi\mathcal{S}^{\varepsilon} &= \pi\{\mathbf{S} \in \Phi_n \,|\, \exists \mathbf{R} \in \mathcal{S} : \mathbf{S} \approx_{\varepsilon} \mathbf{R}\} \\
&= \{\pi\mathbf{S} \,|\, \mathbf{S} \in \Phi_n, \exists \mathbf{R} \in \mathcal{S} : \mathbf{S} \approx_{\varepsilon} \mathbf{R}\} \\
&\subseteq \{\pi\mathbf{S} \,|\, \mathbf{S} \in \Phi_n, \exists \mathbf{R} \in \mathcal{S} : \pi\mathbf{S} \approx_{\varepsilon^{\pi}} \pi\mathbf{R}\} \\
&= \{\mathbf{T} \in \pi\Phi_n \,|\, \exists \mathbf{U} \in \pi\mathcal{S} : \mathbf{T} \approx_{\varepsilon^{\pi}} \mathbf{U}\} \\
&\subseteq \{\mathbf{T} \in \Phi_n \,|\, \exists \mathbf{U} \in \pi\mathcal{S} : \mathbf{T} \approx_{\varepsilon^{\pi}} \mathbf{U}\} \\
&= (\pi\mathcal{S})^{\varepsilon^{\pi}}
\end{aligned}$$

  where the third step follows from the fact that $\mathbf{S} \approx_{\varepsilon} \mathbf{R} \implies \pi\mathbf{S} \approx_{\varepsilon^{\pi}} \pi\mathbf{R}$, which is true because $\Delta^{\mathbf{D}}(\pi\mathbf{S}, \pi\mathbf{R}) = \Delta^{\mathbf{D}\pi}(\mathbf{S}, \mathbf{R}) \leq \varepsilon(\mathbf{D}\pi)$, for any $\mathbf{D} \in \Theta_n$.

Together with (2), the above observations directly imply that the anonymous construction statements compose in the following way:

$$\mathcal{R}^{*_{\mathcal{C}}} \xrightarrow{\pi_1^{\overline{\mathcal{C}}}} \left(\mathcal{S}^{*_{\mathcal{C}\cup\{E\}}}\right)^{\varepsilon_1} \;\wedge\; \mathcal{S}^{*_{\mathcal{C}}} \xrightarrow{\pi_2^{\overline{\mathcal{C}}}} \left(\mathcal{T}^{*_{\mathcal{C}\cup\{E\}}}\right)^{\varepsilon_2}$$

$$\implies \quad \mathcal{R}^{*_{\mathcal{C}}} \xrightarrow{\pi_2^{\overline{\mathcal{C}}}\pi_1^{\overline{\mathcal{C}}}} \left(\mathcal{T}^{*_{\mathcal{C}\cup\{E\}}}\right)^{\varepsilon_1^{\pi_2^{\overline{\mathcal{C}}}}+\varepsilon_2},$$

where we used the fact that for any specification $\mathcal{R}$ and any functions $\varepsilon_1, \varepsilon_2$, we have $(\mathcal{R}^{\varepsilon_1})^{\varepsilon_2} = \mathcal{R}^{\varepsilon_1+\varepsilon_2}$, with $(\varepsilon_1 + \varepsilon_2)(\mathbf{D}) \doteq \varepsilon_1(\mathbf{D}) + \varepsilon_2(\mathbf{D})$ [JM20, Theorem 2].

D Relations Among Game-Based Notions

In the following we use the trivial fact that for distinguisher $\mathbf{D}$ and systems $\mathbf{S}_1, \ldots, \mathbf{S}_n$,

$$\Delta^{\mathbf{D}}(\mathbf{S}_1, \mathbf{S}_n) \leq \sum_{i=1}^{n-1} \Delta^{\mathbf{D}}(\mathbf{S}_i, \mathbf{S}_{i+1}),$$

which directly implies

$$\forall i \in [n-1] : \mathbf{S}_i \approx_{\varepsilon_i} \mathbf{S}_{i+1} \quad \implies \quad \mathbf{S}_1 \approx_{\varepsilon} \mathbf{S}_n, \text{ for } \varepsilon(\mathbf{D}) \doteq \sum_{i=1}^{n-1} \varepsilon_i(\mathbf{D}). \tag{3}$$

This is essentially the *hybrid argument*. Moreover, since for any distinguisher $\mathbf{D}$, reduction system $\mathbf{C}$, and systems $\mathbf{S}$ and $\mathbf{T}$ such that $\mathbf{S} \approx_{\varepsilon} \mathbf{T}$,

$$\Delta^{\mathbf{D}}(\mathbf{CS}, \mathbf{CT}) = \Delta^{\mathbf{DC}}(\mathbf{S}, \mathbf{T}) \leq \varepsilon(\mathbf{DC}) \doteq \varepsilon^{\mathbf{C}}(\mathbf{D}),$$

we have

$$\mathbf{S} \approx_{\varepsilon} \mathbf{T} \quad \implies \quad \mathbf{CS} \approx_{\varepsilon^{\mathbf{C}}} \mathbf{CT}. \tag{4}$$

D.1 Bilateral Signatures

In this section we first define separate notions of authenticity (UF security) and anonymity (IK security) for bilateral signatures, and then show that they imply the UF-IK security notion from Definition 4. We further relate the UF and IK notions we introduce here to those from the literature in Appendix E.1.

We begin by defining authenticity of bilateral signatures. For this, we define a distinction problem between a real system that correctly generates and verifies signatures, via a signing oracle for one sender and a verification oracle for one receiver, and an ideal system that correctly generates signatures, but only correctly verifies signatures previously output by the signing oracle.

Definition 13 (UF-Secure Bilateral Signature). *A bilateral signature scheme* Σ_{BS} *is* ε-unforgeable *(or* ε-UF-secure*) if*

$$[\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}_{rsk,spk}\rangle, \mathbf{K}_{spk,rpk}] \approx_\varepsilon [\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}^{\perp}\rangle, \mathbf{K}_{spk,rpk}],$$

for key-pairs $(ssk, spk) \leftarrow \mathtt{Gen}_S$ *and* $(rsk, rpk) \leftarrow \mathtt{Gen}_R$*.*

Note that usually when authenticity is interpreted as unforgeability, as we do here, the related security notion is defined as a game where an adversary must first interact with a system implementing some oracles, and eventually attempt to come up with a concrete forgery. Nevertheless, defining unforgeability (hence, authenticity) through a distinction problem is not uncommon (see [Ros18] for example). The latter suits us better because it more directly relates to composable notions of security, and moreover it can be easily shown that it is implied by the former: as opposed to the real system, valid forgeries in the ideal system are falsely reported to be incorrect, thus trivially allowing to distinguish (see Appendix C.2 of [BM20] for a more detailed discussion).

We next define anonymity of bilateral signatures. For this, we define a distinction problem between a real system that correctly generates and verifies signatures, via signing and verification oracles for n (different) senders and one receiver, and an ideal system that also correctly generates and verifies signatures, but via n copies of signing and verification oracles for the *same* (randomly selected) sender and one receiver.

Definition 14 (IK-Secure Bilateral Signature). *A bilateral signature scheme* Σ_{BS} *is* (n, ε)-anonymous *(or* (n, ε)-IK-secure*) if*

$$\begin{gathered}[\langle \mathbf{S}_{ssk_1,rpk}, \mathbf{V}_{rsk,spk_1}\rangle, \ldots, \langle \mathbf{S}_{ssk_n,rpk}, \mathbf{V}_{rsk,spk_n}\rangle, \mathbf{K}_{\boldsymbol{spk},rpk}] \\ \approx_\varepsilon \\ \underbrace{[\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I}\rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}],\end{gathered}$$

for key-pairs $(ssk_1, spk_1), \ldots, (ssk_n, spk_n) \leftarrow \mathtt{Gen}_S$*,* $(rsk, rpk) \leftarrow \mathtt{Gen}_R$*,* $\boldsymbol{spk} \doteq (spk_1, \ldots, spk_n)$*, random variable* $I \xleftarrow{\$} [n]$*, and where both systems are such that if a signature obtained from the* i*-th signing oracle is input to the* j*-th verification oracle, for* $j \neq i$*, then* $\perp$ *is output.*

We now show that these two notions together imply our combined notion from Definition 4.

Lemma 3. *There exists a reduction system* $\mathbf{C}$ *such that, if a bilateral signature scheme* Σ_{BS} *is* ε-UF*-secure and* (n, ε')-IK*-secure, then it is* $(n, \varepsilon^{\mathbf{C}} + \varepsilon')$-$\mathsf{UF}$-$\mathsf{IK}$-*secure.*

Proof. By assumption, recalling Definitions 13 and 14, we have

$$[\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}_{rsk,spk} \rangle, \mathbf{K}_{spk,rpk}] \approx_{\varepsilon} [\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{spk,rpk}]$$

and

$$[\langle \mathbf{S}_{ssk_1,rpk}, \mathbf{V}_{rsk,spk_1} \rangle, \ldots, \langle \mathbf{S}_{ssk_n,rpk}, \mathbf{V}_{rsk,spk_n} \rangle, \mathbf{K}_{\boldsymbol{spk},rpk}]$$
$$\approx_{\varepsilon'}$$
$$[\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}].$$

We can now easily define a reduction system $\mathbf{C}$ such that

$$\mathbf{C}[\langle \mathbf{S}, \mathbf{V} \rangle, \mathbf{K}_{spk,rpk}] = [\underbrace{\langle \mathbf{S}, \mathbf{V} \rangle, \ldots, \langle \mathbf{S}, \mathbf{V} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}],$$

for $\langle \mathbf{S}, \mathbf{V} \rangle \in \{\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}_{rsk,spk} \rangle, \langle \mathbf{S}_{ssk,rpk}, \mathbf{V}^{\perp} \rangle\}$. $\mathbf{C}$ simply emulates n copies of $\langle \mathbf{S}, \mathbf{V} \rangle$, and samples $n-1$ key-pairs and an index $I \overset{\$}{\leftarrow} [n]$, and then emulates $\mathbf{K}_{\boldsymbol{spk},rpk}$ by constructing $\boldsymbol{spk}$ such that $spk_I = spk$ and using the other public keys from the sampled key-pairs. This way, we clearly have that

$$\mathbf{C}[\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}_{rsk,spk} \rangle, \mathbf{K}_{spk,rpk}]$$
$$= [\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}]$$

and

$$\mathbf{C}[\langle \mathbf{S}_{ssk,rpk}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{spk,rpk}] = [\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}].$$

Then by (4) we have

$$[\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}_{rsk,spk_I} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}]$$
$$\approx_{\varepsilon^{\mathbf{C}}}$$
$$[\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}].$$

Finally, by (3) we have

$$[\langle \mathbf{S}_{ssk_1,rpk}, \mathbf{V}_{rsk,spk_1} \rangle, \ldots, \langle \mathbf{S}_{ssk_n,rpk}, \mathbf{V}_{rsk,spk_n} \rangle, \mathbf{K}_{\boldsymbol{spk},rpk}]$$
$$\approx_{\varepsilon^{\mathbf{C}}+\varepsilon'}$$
$$[\underbrace{\langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{ssk_I,rpk}, \mathbf{V}^{\perp} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk},rpk}],$$

which according to Definition 4 concludes the proof. □

D.2 Partial Signatures

In this section we first define separate notions of authenticity (UF security) and unambiguity (UA security) for partial signatures, and then show that they imply the UF-UA security notion from Definition 7. We further relate the UF and UA notions we introduce here to those from the literature in Appendix E.2.

We begin by defining authenticity of partial signatures. For this, we define a distinction problem between a real system that correctly generates and verifies signatures (stub-tag pairs) and an ideal system that correctly generates signatures, but only correctly verifies signatures previously output by the signing oracle.

Definition 15 (UF-Secure Partial Signature). *A partial signature scheme Σ_{PS} is* ε-unforgeable *(or* ε-UF-secure*) if*

$$[\langle \mathbf{S}_{sk}, \mathbf{V}_{pk} \rangle, \mathbf{K}_{pk}] \approx_{\varepsilon} [\langle \mathbf{S}_{sk}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{pk}],$$

for key-pair $(sk, pk) \leftarrow \mathtt{Gen}$.

We next define unambiguity of partial signatures. For this, we first need to define the following additional systems (where the dependency on a fixed Σ_{PS} scheme is implicit), parameterized by keys $sk_1, \ldots, sk_n \in \mathcal{SK}$ and $pk_1, \ldots, pk_n \in \mathcal{PK}$, for any $n \in \mathbb{N}$. Here, we will additionally assume that such systems implicitly share state with other systems that are identical up to the sampled key-pair.

– $\langle \mathbf{S}_{sk_i}, \mathbf{V}^{\perp}_{pk_i} \rangle$: Set the *shared* set $\mathcal{Q}_i \subseteq \mathcal{S}$ to $\varnothing$ and then:
  - On input $m \in \mathcal{M}$, return $(m, \sigma, \tau) \in \mathcal{M} \times \mathcal{S} \times \mathcal{T}$, for $(\sigma, \tau) \leftarrow \mathtt{Sgn}_{sk}(m)$, and set $\mathcal{Q}_i$ to $\mathcal{Q}_i \cup \{\sigma\}$.
  - On input $(m, \sigma, \tau) \in \mathcal{M} \times \mathcal{S} \times \mathcal{T}$, return m if $\sigma \in \mathcal{Q}_i$ and $\perp$ otherwise.

We can now define a distinction problem between a real system that correctly generates and verifies signatures, via signing and verification oracles for n (different) senders, and an ideal system that also correctly generates signatures for n (different) senders, but for each only correctly verifies signatures previously signed by its associated signing oracle. This property can be thought of as the analogue of robustness of PKE from [ABN10] for (partial) signatures.

Definition 16 (UA-Secure Partial Signature). *A partial signature scheme Σ_{PS} is* (n, ε)-unambiguous *(or* (n, ε)-UA-secure*) if*

$$[\langle \mathbf{S}_{sk_1}, \mathbf{V}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \approx_{\varepsilon} [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}],$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \mathtt{Gen}$ and $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$.

We now show that these two notions together imply our combined notion from Definition 7.

Lemma 4. *There exists a reduction system* $\mathbf{C}$ *such that, if a partial signature scheme* Σ_{PS} *is* ε-UF*-secure and* (n, ε')-UA*-secure, then it is* $(n, n \cdot \varepsilon^{\mathbf{C}} + \varepsilon')$-$\mathsf{UF}$-$\mathsf{UA}$*-secure.*

Proof. By assumption, recalling Definitions 15 and 16, we have

$$[\langle \mathbf{S}_{sk}, \mathbf{V}_{pk} \rangle, \mathbf{K}_{pk}] \approx_{\varepsilon} [\langle \mathbf{S}_{sk}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{pk}],$$

and

$$[\langle \mathbf{S}_{sk_1}, \mathbf{V}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \approx_{\varepsilon'} [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}].$$

We can now easily define a reduction system $\mathbf{C}$ such that

$$\begin{aligned}\mathbf{C}[\langle \mathbf{S}, \mathbf{V} \rangle, \mathbf{K}_{pk}] = [&\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_{I-1}}, \mathbf{V}^{\perp}_{pk_{I-1}} \rangle, \langle \mathbf{S}, \mathbf{V} \rangle, \\ &\langle \mathbf{S}_{sk_{I+1}}, \mathbf{V}^{\perp}_{pk_{I+1}} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp}_{pk_n} \rangle, \mathbf{K}_{(pk_1, \ldots, pk_{I-1}, pk, pk_{I+1}, \ldots, pk_n)}],\end{aligned}$$

for $\langle \mathbf{S}, \mathbf{V} \rangle \in \{\langle \mathbf{S}_{sk}, \mathbf{V}_{pk} \rangle, \langle \mathbf{S}_{sk}, \mathbf{V}^{\perp} \rangle\}$ and $I \xleftarrow{\$} [n]$, and where $\mathbf{C}$ also keeps an appropriate set $\mathcal{Q}_I$ for the verification oracle $\mathbf{V}$. This way, we clearly have that

$$\mathbf{C}[\langle \mathbf{S}_{sk}, \mathbf{V}_{pk} \rangle, \mathbf{K}_{pk}] = [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}]$$

and

$$\mathbf{C}[\langle \mathbf{S}_{sk}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{pk}] = [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{\boldsymbol{pk}}].$$

Then by (4) and the union bound we have

$$\begin{gathered}[\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \\ \approx_{n \cdot \varepsilon^{\mathbf{C}}} \\ [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{\boldsymbol{pk}}].\end{gathered}$$

Finally, by (3) we have

$$\begin{gathered}[\langle \mathbf{S}_{sk_1}, \mathbf{V}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \\ \approx_{n \cdot \varepsilon^{\mathbf{C}} + \varepsilon'} \\ [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{\boldsymbol{pk}}],\end{gathered}$$

which according to Definition 7 concludes the proof. □

D.3 Ring Signatures

We begin by defining authenticity of ring signatures. For this, we first need to define the following additional systems (where the dependency on a fixed Σ_{RS} scheme is implicit), parameterized by keys $\boldsymbol{sk} \doteq (sk_1, \ldots, sk_n) \in \mathcal{SK}^n$ and $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n) \in \mathcal{PK}^n$, for any $n \in \mathbb{N}$.

– $\langle \mathbf{S}_{\boldsymbol{sk}, \boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle$:

  - On input $(i, m) \in [n] \times \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $\sigma \leftarrow \mathtt{Sgn}_{i,sk,\boldsymbol{pk}}(m)$.
  - On input $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, return m if $\mathtt{Vrf}_{\boldsymbol{pk}}(m, \sigma) = 1$ and $\perp$ otherwise.
- $\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle$: Set $\mathcal{Q} \subseteq \mathcal{M} \times \mathcal{S}$ to $\varnothing$, and then:
  - On input $(i, m) \in [n] \times \mathcal{M}$, return $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, for $\sigma \leftarrow \mathtt{Sgn}_{i,sk,\boldsymbol{pk}}(m)$, and set $\mathcal{Q}$ to $\mathcal{Q} \cup \{(m, \sigma)\}$.
  - On input $(m, \sigma) \in \mathcal{M} \times \mathcal{S}$, return m if $(m, \sigma) \in \mathcal{Q}$ and $\perp$ otherwise.

We can now define a distinction problem between a real system that correctly generates and verifies signatures, via a signing oracle for one sender and a verification oracle for one receiver, and an ideal system that correctly generates signatures, but only correctly verifies signatures previously output by the signing oracle.

Definition 17 (UF-Secure Ring Signature). *A ring signature scheme Σ_{RS} is (n, ε)-*unforgeable *(or (n, ε)-*UF-secure*) if*

$$[\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \approx_{\varepsilon} [\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{\boldsymbol{pk}}],$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow$ `Gen`, $\boldsymbol{sk} \doteq (sk_1, \ldots, sk_n)$, and $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$.

We next define anonymity of ring signatures. For this, we define a distinction problem between a real system that correctly generates and verifies signatures, via signing and verification oracles for n (different) senders, and an ideal system that also correctly generates and verifies signatures, but via n copies of signing and verification oracles for the *same* (randomly selected) sender.

Definition 18 (IK-Secure Ring Signature). *A ring signature scheme Σ_{RS} is (n, ε)-*anonymous *(or (n, ε)-*IK-secure*) if*

$$\begin{gathered}[\langle \mathbf{S}_{1,sk_1,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \ldots, \langle \mathbf{S}_{n,sk_n,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \mathbf{K}_{\boldsymbol{pk}}] \\ \approx_{\varepsilon} \\ [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \ldots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}],\end{gathered}$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow$ `Gen`, $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$, random variable $I \xleftarrow{\$} [n]$, and where both systems are such that if a signature obtained from the i-th signing oracle is input to the j-th verification oracle, for $j \neq i$, then $\perp$ is output.

We now show that these two notions together imply our combined notion from Definition 11.

Lemma 5. *There exists a reduction system $\mathbf{C}$ such that, if a ring signature scheme Σ_{RS} is (n, ε)-*UF*-secure and (n, ε')-*IK*-secure, then it is $(n, \varepsilon^{\mathbf{C}} + \varepsilon')$-*UF*-*IK*-secure.*

Proof. By assumption, recalling Definitions 17 and 18, we have

$$[\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \mathbf{K}_{\boldsymbol{pk}}] \approx_{\varepsilon} [\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle, \mathbf{K}_{\boldsymbol{pk}}]$$

and

$$\begin{gathered}[\langle \mathbf{S}_{1,sk_1,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \dots, \langle \mathbf{S}_{n,sk_n,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \mathbf{K}_{\boldsymbol{pk}}] \\ \approx_{\varepsilon'} \\ [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \dots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}].\end{gathered}$$

We can now easily define a reduction system $\mathbf{C}$ such that

$$\mathbf{C}[\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}\rangle, \mathbf{K}_{\boldsymbol{pk}}] = [\underbrace{\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}(I,\cdot), \mathbf{V}\rangle, \dots, \langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}(I,\cdot), \mathbf{V}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}],$$

for $I \xleftarrow{\$} [n]$, $\mathbf{V} \in \{\mathbf{V}_{\boldsymbol{pk}}, \mathbf{V}^{\perp}\}$, and where $\mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}(i,\cdot)$, for $i \in [n]$, is the system $\mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}$ where the first argument of the queried tuple is fixed to i. This way, we clearly have that

$$\mathbf{C}[\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \mathbf{K}_{\boldsymbol{pk}}] = [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \dots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}]$$

and

$$\mathbf{C}[\langle \mathbf{S}_{\boldsymbol{sk},\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle, \mathbf{K}_{\boldsymbol{pk}}] = [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle, \dots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}].$$

Then by (4) we have

$$\begin{gathered}[\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \dots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}] \\ \approx_{\varepsilon^{\mathbf{C}}} \\ [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle, \dots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}].\end{gathered}$$

Finally, by (3) we have

$$\begin{gathered}[\langle \mathbf{S}_{1,sk_1,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \dots, \langle \mathbf{S}_{n,sk_n,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}}\rangle, \mathbf{K}_{\boldsymbol{pk}}] \\ \approx_{\varepsilon^{\mathbf{C}}+\varepsilon'} \\ [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle, \dots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp}\rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}],\end{gathered}$$

which according to Definition 11 concludes the proof. □

E Relations with Previous Notions and Schemes

In this section we relate our new game-based notions to the ones from the literature. We also discuss known schemes achieving them, which therefore also achieve our composable notions.

E.1 Bilateral Signatures

As we pointed out earlier, bilateral signatures share the same syntax of designated verifier signatures (DVS). This does not mean that, as a cryptographic scheme, they are the same. In fact, what matters are also the *semantics* of such scheme, that is, how its security is defined. On a high level, for DVS (game-based) security corresponds to being unable to tell whether a signature was produced by the sender or by the receiver, and therefore anonymity in not necessarily guaranteed among signatures generated by different senders. Instead, for bilateral signatures, the latter property is exactly what defines security, in terms of anonymity. Moreover, the characterizing feature of DVS is irrelevant: For bilateral signatures, we do not want to (necessarily) hide the role of the sender, or respectively of the receiver; a bilateral signature scheme in principle allows an adversary to tell that the signature was generated by one of the senders, and in particular, *not* by the receiver, and therefore such a scheme would *not* be a secure DVS. Recently, in [MPR21] this characterizing feature of DVS that hides *both* the sender and the receivers has been modeled composably, where guarantees are provided not only to honest parties, but also to dishonest ones.

Nevertheless, in [JSI96], where DVS were originally introduced, the concept of *strong* DVS was mentioned, requiring a DVS scheme to additionally provide indistinguishably of signatures produced by different senders (the same property capturing anonymity of bilateral signatures). This notion was later formalized in [LV05], and it was shown how to enhance any DVS scheme to additionally satisfy this stronger notion, dubbed PSI-CMA-security. Clearly, such a DVS scheme would also be a bilateral signature scheme, albeit not *minimal*, in the sense that it would provide additional unnecessary security guarantees.

We now informally argue that the concrete scheme DVSBMH from [LV05] achieves our composable notion for bilateral signatures, that is, it constructs $\mathsf{A\text{-}AUT}_{n\to 1}$ from $[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{1\text{-}AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}]$ when used in the natural way. To do so, it suffices to relate the notions DVSBMH has been shown to satisfy to our game-based notions of UF-security and IK-security; then Lemma 3 implies that DVSBMH is also UF-IK-secure, and by Theorem 1 it is therefore also composably secure, as per Definition 5. Note that, syntactically, DVSBMH is actually a *universal* DVS (UDVS) scheme, that is, a regular signature scheme equipped with additional functions emulating those of a DVS scheme. Therefore, using DVSBMH in the natural way means in particular to first produce a signature with the base signing function, and then feeding it along with the message and the receiver's public key to a further "designation" function, which will produce the final signature to be transmitted.

Unforgeability. In [LV05] DVSBMH has been shown to be ST-DV-UF-secure, a notion introduced in [SWP04] which is a stronger version of the earlier notion of DV-UF-security from [SBWP03]. The former is stronger in the sense that, unlike the latter, it provides the attacker access to the verification oracle (in addition to a signing one), and therefore it directly relates to our UF-security notion for bilateral signatures from Definition 13.

Anonymity. In [LV05] DVSBMH has been shown to be $\mathsf{PSI\text{-}CMA}$-secure, a notion introduced there and that also relates to our counterpart for bilateral signatures, IK-security from Definition 14, but less directly. This is because $\mathsf{PSI\text{-}CMA}$-security is essentially defined as key-indistinguishability of signatures, but only for *two* senders and one receiver, and therefore the IK-security of DVSBMH incurs a loss of multiplicative factor $(n-1)$, which can be shown via a standard hybrid argument.

E.2 Partial Signatures

Our game-based definitions for partial signatures closely resemble the ones from the literature, except that we chose to phrase the notions as distinction problems, whereas [BD09] defines unforgeability and unambiguity as forgery problems and anonymity as a bit-guessing problem. [BD09] also introduces various constructions satisfying their definitions, one being the so-called StC (sign-then-commit) construction. This partial signature scheme is based on a regular signature scheme and a commitment scheme, and works as follows: to create a stub-tag pair (σ, τ) on a message m under secret-key sk (and corresponding public-key pk), the new signing function simply produces a regular signature s on m using the base signature scheme, then produces a commitment-decommitment pair (c, d) on the concatenation of s and pk, and finally sets $\sigma \doteq c$ and $\tau \doteq (s, d)$. Verification is then defined in the straightforward way.

We now informally argue that the simple StC construction[4] achieves our composable notion for bilateral signatures, that is, it constructs $\mathsf{D\text{-}AUT}_{n\to 1}$ from $[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{A\text{-}INS}_{n\to 1}]$ when used in the natural way. To do so, it suffices to relate the notions StC has been shown to satisfy to our game-based notions of UF-security, UA-security, and IK-security; then Lemma 4 implies that StC is also $\mathsf{UF\text{-}UA}$-secure, and by Theorem 2 it is therefore also composably secure, as per Definition 9.

Unforgeability. In [BD09] StC has been shown to be unforgeable if the base signature scheme is unforgeable and the base commitment scheme is hiding. The unforgeability notion for partial signatures from [BD09] is slightly stronger than ours, in the sense that the signing oracle only returns stubs, and allows the adversary to later selectively see any associated tags. Such notion can be appropriately weakened, and then shown to be equivalent to our distinction problem from Definition 15, since being able to distinguish the two systems implies being able to find a valid forgery. Therefore, StC also satisfies our UF-security notion for partial signatures.

Unambiguity. In [BD09] StC has been shown to be unambiguous if the base commitment scheme is binding. The unforgeability notion for partial signatures from [BD09] is slightly stronger than ours, in the sense that the adversary can choose itself public keys, messages, stub and tags of the forgery. Such notion can

[4] One could make analogous arguments for the other constructions from [BD09].

be appropriately weakened, and then shown to be equivalent to our distinction problem from Definition 16, since being able to distinguish the two systems implies being able to find a valid forgery. In more detail, this is so because the two systems behave identically until the distinguisher manages to come up with a verification query (m', σ, τ') for the j-th verification oracle such that it previously queried the i-th signing oracle on m, for $i \neq j$, and obtained (σ, τ), and hence distinguishing between the two implies finding such a forgery. Therefore, StC also satisfies our UA-security notion for partial signatures.

Anonymity. In [BD09] StC has been shown to be anonymous if the base commitment scheme is hiding. The anonymity notion for partial signatures from [BD09] is slightly different than ours because it is only defined for two senders, and it is phrased as a bit-guessing problem. Nevertheless, it can be shown to be equivalent to our distinction problem from Definition 8, up to a multiplicative loss factor of $(n-1)$, via a standard hybrid argument. Therefore, StC also satisfies our IK-security notion for partial signatures.

E.3 Ring Signatures

Our game-based definitions for ring signatures closely resemble the ones from the literature, except that we chose to phrase the notions as distinction problems, whereas [BKM06] defines unforgeability as a forgery problem and anonymity as a bit-guessing problem. [BKM06] also introduces a construction satisfying their (stronger) definitions, which we call the BKM construction here. This ring signature scheme is based on a public-key encryption scheme, a regular signature scheme, a ZAP (i.e., a two-round public-coin witness-indistinguishable proof system, where the first round is a random string from the verifier to the prover), and roughly works as follows: Sender S_i initially generates a public-key encryption key-pair (sk_i^E, pk_i^E) and a regular signature key-pair (sk_i^S, pk_i^S). In order to generate a ring signature on a message m, S_i first produces a regular signature σ' on m with its signing key sk_i^S. Then S_i produces ciphertexts C_j^*, for $j \in [n]$, using encryption keys $pk_1^S, \ldots, pk_n^S$, where C_i^* is the encryption of σ' and the other ciphertexts are encryptions of random bit-strings instead. Finally, using the ZAP S_i produces a proof π, stating that one of the ciphertexts is indeed an encryption of a valid signature on m with respect to the public verification key of one of the ring members (that is, pk_i^S). Verification is then defined in the straightforward way.

We now informally argue that the BKM construction achieves our composable notion for ring signatures, that is, it constructs $\mathsf{RA\text{-}AUT}_{n\to 1}$ from $[\mathsf{1\text{-}AUT}_{n \circlearrowright 1}, \mathsf{A\text{-}INS}_{n\to 1}]$ when used in the natural way. To do so, we first observe that the stronger notions of *unforgeability w.r.t. insider corruption* and *anonymity against attribution attacks* that BKM has been shown to satisfy in [BKM06], trivially imply the weaker notions of *unforgeability against fixed-ring attacks* and *basic anonymity*, respectively, that [BKM06] also defines. It then suffices to relate the latter notions to our game-based notions of UF-security and IK-security, respectively, since Lemma 5 then implies that BKM is also

UF-IK-secure, which by Theorem 3 is therefore also composably secure, as per Definition 12.

Unforgeability. In [BKM06] the BKM construction has been shown to be unforgeable against insider corruption, and therefore also against fixed-ring attacks, if the base signature scheme is unforgeable. The original notion of unforgeability against fixed-ring attacks, [BKM06, Definition 5], can be shown to be equivalent to our distinction problem from Definition 15, since being able to distinguish the two systems implies being able to find a valid forgery. Therefore, the BKM construction also satisfies our UF-security notion for ring signatures.

Anonymity. In [BKM06] the BKM construction has been shown to be anonymous against attribution attacks, and therefore it also satisfies basic anonymity, if the base public-key encryption scheme is IND-CPA-secure and the ZAP is witness-indistinguishable. The original notion of basic anonymity, [BKM06, Definition 5], is slightly different than our IK-security notion because it is only defined for two senders, and it is phrased as a bit-guessing problem. Nevertheless, it can be shown to be equivalent to our distinction problem from Definition 18, up to a multiplicative loss factor of $(n-1)$, via a standard hybrid argument. Therefore, the BKM construction also satisfies our IK-security notion for ring signatures.

F Proofs of Main Results

To prove anonymous construction statements, we will rely on the following lemma.

Lemma 6. *For an n-protocol π, a function ε, and n-resources $\mathbf{R}, \mathbf{S}$, if there exists a simulator $\mathsf{sim} \in \Sigma$ such that $\pi\mathbf{R} \approx_\varepsilon \mathsf{sim}^E\, \mathbf{S}$, then $\mathbf{R} \xmapsto{\pi,\varepsilon} \mathbf{S}$.*

Proof. Since by definition $\pi\mathbf{R} \in (\mathsf{sim}^E\, \mathbf{S})^\varepsilon \subseteq (\mathbf{S}^{*_E})^\varepsilon$, we have $\{\pi\mathbf{R}\} \subseteq (\mathbf{S}^{*_E})^\varepsilon$. Then it clearly follows that $\pi^{\overline{\mathcal{C}}}\, \mathbf{R}^{*_\mathcal{C}} \subseteq (\mathbf{S}^{*_{\mathcal{C} \cup \{E\}}})^\varepsilon$ for any $\mathcal{C} \subseteq \{S_i\}_{i=1}^n$. □

F.1 Anonymous Authenticity

Theorem 1. *There exists a reduction system $\mathbf{C}$ such that, if a bilateral signature scheme Σ_{BS} is (n, ε)-UF-IK-secure, then it is $(n, \varepsilon^{\mathbf{C}})$-composably secure.*

Proof. Let define systems

$$\mathbf{R} \doteq [\langle \mathbf{S}_{ssk_1, rpk}, \mathbf{V}_{rsk, spk_1} \rangle, \ldots, \langle \mathbf{S}_{ssk_n, rpk}, \mathbf{V}_{rsk, spk_n} \rangle, \mathbf{K}_{\boldsymbol{spk}, rpk}],$$
$$\mathbf{S} \doteq [\underbrace{\langle \mathbf{S}_{ssk_I, rpk}, \mathbf{V}^\perp \rangle, \ldots, \langle \mathbf{S}_{ssk_I, rpk}, \mathbf{V}^\perp \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{spk}, rpk}],$$

for key-pairs $(ssk_1, spk_1), \ldots, (ssk_n, spk_n) \leftarrow \mathtt{Gen}_S$, $(rsk, rpk) \leftarrow \mathtt{Gen}_R$, and random variable $I \xleftarrow{\$} [n]$. Then by Definition 4, Σ_{BS} is such that $\mathbf{R} \approx_\varepsilon \mathbf{S}$. We now need to provide a simulator sim and a reduction system $\mathbf{C}$ such that

$$\mathbf{CR} = \pi_{\mathsf{BS}}[\text{1-AUT}_{n\to 1}, \text{1-AUT}_{n\leftarrow 1}, \text{A-INS}_{n\to 1}],$$
$$\mathbf{CS} = \mathsf{sim}^E\, \text{A-AUT}_{n\to 1}.$$

This way we have $\Delta^{\mathbf{DC}}(\mathbf{R},\mathbf{S}) \leq \varepsilon(\mathbf{DC}) = \varepsilon^{\mathbf{C}}(\mathbf{D})$, for any distinguisher $\mathbf{D}$, and therefore

$$\Delta^{\mathbf{D}}(\pi_{\mathsf{BS}}[\text{1-}\mathsf{AUT}_{n\to 1}, \text{1-}\mathsf{AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}], \mathsf{sim}^{E}\, \mathsf{A\text{-}AUT}_{n\to 1}) \leq \varepsilon^{\mathbf{C}}(\mathbf{D}),$$

which by Lemma 6 proves the theorem.

The simulator sim first sets $\mathcal{Q} \leftarrow \varnothing$ and samples a random index $I \overset{\$}{\leftarrow} [n]$. Then it generates n sender key-pairs $(ssk_1, spk_1), \ldots, (ssk_n, spk_n) \leftarrow \mathtt{Gen}_S$ as well as one receiver key-pair $(rsk, rpk) \leftarrow \mathtt{Gen}_R$, and once the adversary inputs $\diamond$ to the interfaces E emulating $\text{1-}\mathsf{AUT}_{n\to 1}$ and $\text{1-}\mathsf{AUT}_{n\leftarrow 1}$, sim outputs $\{(i, spk_i) \mid i \in [n]\}$ and rpk, respectively, at the same interface. Whenever the adversary inputs $\diamond$ to the interfaces E emulating $\mathsf{A\text{-}INS}_{n\to 1}$, sim also inputs $\diamond$ to the interface E of $\mathsf{A\text{-}AUT}_{n\to 1}$, obtaining a set $\mathfrak{O} \subseteq \mathbb{N} \times \mathcal{M}$. It then outputs the set $\{(j, m, \mathtt{Sgn}_{ssk_I, rpk}(m)) \mid \exists\, (j, m) \in \mathfrak{O}\}$ to E, and sets $\mathcal{Q} \leftarrow \mathcal{Q} \cup \mathfrak{O}$. Whenever the adversary inputs (m, σ) to the interface E emulating $\mathsf{A\text{-}INS}_{n\to 1}$, if $(j, m, \sigma) \in \mathcal{Q}$ for some $j \in \mathbb{N}$, then sim inputs j to the E interface of $\mathsf{A\text{-}AUT}_{n\to 1}$.

The reduction system $\mathbf{C}$ interacts with a system $[\langle \mathbf{S}_1, \mathbf{V}_1 \rangle, \ldots, \langle \mathbf{S}_n, \mathbf{V}_n \rangle, \mathbf{K}]$, which is either $\mathbf{R}$ or $\mathbf{S}$. $\mathbf{C}$ works by emulating $\pi_{\mathsf{BS}}[\text{1-}\mathsf{AUT}_{n\to 1}, \text{1-}\mathsf{AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}]$, but replacing any call to $\mathtt{Gen}_R, \mathtt{Gen}_S$ by $\mathbf{K}$, any call to $\mathtt{Sgn}_{ssk_i, rpk}$ by $\mathbf{S}_i$, and any call to $\mathtt{Vrf}_{rsk, spk_i}$ by $\mathbf{V}_i$. Then clearly $\mathbf{CR} = \pi_{\mathsf{BS}}[\text{1-}\mathsf{AUT}_{n\to 1}, \text{1-}\mathsf{AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}]$, and it is also easy to see that $\mathbf{CS} = \mathsf{sim}^{E}\, \mathsf{A\text{-}AUT}_{n\to 1}$. □

F.2 De-Anonymizable Authenticity

Theorem 2. *There exist reduction systems* $\mathbf{C}_{\mathsf{m}}$ *and* $\mathbf{C}_{\mathsf{h}}$ *such that, if a partial signature scheme* Σ_{PS} *is* $(n, \varepsilon_{\mathsf{m}})$*-*$\mathsf{IK}$*-secure and* $(n, \varepsilon_{\mathsf{h}})$*-*$\mathsf{UF\text{-}UA}$*-secure, then it is* $(n, t, \varepsilon_{\mathsf{m}}^{\mathbf{C}_{\mathsf{m}}}, \varepsilon_{\mathsf{h}}^{\mathbf{C}_{\mathsf{h}}})$*-composably secure, for any* $t \in \mathbb{N}$.

Proof. Let $t \in \mathbb{N}$ and define systems

$$\begin{aligned}
\mathbf{R}_{\mathsf{m}} &\doteq [\mathbf{S}^{-}_{sk_1}, \ldots, \mathbf{S}^{-}_{sk_n}, \mathbf{K}_{\boldsymbol{pk}}],\\
\mathbf{S}_{\mathsf{m}} &\doteq [\underbrace{\mathbf{S}^{-}_{sk_I}, \ldots, \mathbf{S}^{-}_{sk_I}}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}],\\
\mathbf{R}_{\mathsf{h}} &\doteq [\langle \mathbf{S}_{sk_1}, \mathbf{V}_{pk_1} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}_{pk_n} \rangle, \mathbf{K}_{\boldsymbol{pk}}],\\
\mathbf{S}_{\mathsf{h}} &\doteq [\langle \mathbf{S}_{sk_1}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{sk_n}, \mathbf{V}^{\perp} \rangle, \mathbf{K}_{\boldsymbol{pk}}],
\end{aligned}$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \mathtt{Gen}$, $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$, and random variable $I \overset{\$}{\leftarrow} [n]$. Then by Definitions 8 and 7, Σ_{PS} is such that $\mathbf{R}_{\mathsf{m}} \approx_{\varepsilon_{\mathsf{m}}} \mathbf{S}_{\mathsf{m}}$ and $\mathbf{R}_{\mathsf{h}} \approx_{\varepsilon_{\mathsf{h}}} \mathbf{S}_{\mathsf{h}}$. We now need to provide simulators $\mathsf{sim}_{\mathsf{h}}$, $\mathsf{sim}_{\mathsf{m}}$ and reduction systems $\mathbf{C}_{\mathsf{h}}$, $\mathbf{C}_{\mathsf{m}}$ so that during interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$, for any $j \in [t]$ such that $P_{\mathsf{msg}(j)}$,

$$\begin{aligned}
\mathbf{C}_{\mathsf{m}}\, \mathbf{R}_{\mathsf{m}} &= \pi_{\mathsf{PS}}[\text{1-}\mathsf{AUT}_{n\to 1}, \mathsf{A\text{-}INS}_{n\to 1}],\\
\mathbf{C}_{\mathsf{m}}\, \mathbf{S}_{\mathsf{m}} &= \mathsf{sim}_{\mathsf{m}}^{E}\, \mathsf{D\text{-}AUT}_{n\to 1},
\end{aligned}$$

and during interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$, for any $j \in [t]$ such that $P_{\mathsf{hnd}(j)}$,

$$\mathbf{C}_{\mathsf{h}}\,\mathbf{R}_{\mathsf{h}} = \pi_{\mathsf{BS}}[\mathsf{1\text{-}AUT}_{n\to1}, \mathsf{A\text{-}INS}_{n\to1}],$$
$$\mathbf{C}_{\mathsf{h}}\,\mathbf{S}_{\mathsf{h}} = \mathsf{sim}_{\mathsf{h}}^{E}\,\mathsf{D\text{-}AUT}_{n\to1}.$$

This way we have

$$\Delta^{\mathbf{D}\mathbf{C}_{\mathsf{m}}}(\mathbf{R}_{\mathsf{m}}, \mathbf{S}_{\mathsf{m}}) \le \varepsilon_{\mathsf{m}}(\mathbf{D}\mathbf{C}_{\mathsf{m}}) = \varepsilon_{\mathsf{m}}^{\mathbf{C}_{\mathsf{m}}}(\mathbf{D})$$

and

$$\Delta^{\mathbf{D}\mathbf{C}_{\mathsf{h}}}(\mathbf{R}_{\mathsf{h}}, \mathbf{S}_{\mathsf{h}}) \le \varepsilon_{\mathsf{h}}(\mathbf{D}\mathbf{C}_{\mathsf{h}}) = \varepsilon_{\mathsf{h}}^{\mathbf{C}_{\mathsf{h}}}(\mathbf{D}),$$

for any distinguisher $\mathbf{D}$, and therefore

$$\Delta^{\mathbf{D}}(\pi_{\mathsf{PS}}[\mathsf{1\text{-}AUT}_{n\to1}, \mathsf{A\text{-}INS}_{n\to1}], \mathsf{sim}^{E}\,\mathsf{D\text{-}AUT}_{n\to1}) \le \varepsilon_{\mathsf{m}}^{\mathbf{C}_{\mathsf{m}}}(\mathbf{D}),$$

during interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$ for any $j \in [t]$ such that $P_{\mathsf{msg}(j)}$, and

$$\Delta^{\mathbf{D}}(\pi_{\mathsf{PS}}[\mathsf{1\text{-}AUT}_{n\to1}, \mathsf{A\text{-}INS}_{n\to1}], \mathsf{sim}^{E}\,\mathsf{D\text{-}AUT}_{n\to1}) \le \varepsilon_{\mathsf{m}}^{\mathbf{C}_{\mathsf{m}}}(\mathbf{D}),$$

during interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$ for any $j \in [t]$ such that $P_{\mathsf{hnd}(j)}$. By appropriately adapting Lemma 6 to intervals, this proves the theorem.

For any $j \in [t]$ such that $P_{\mathsf{msg}(j)}$, the simulator $\mathsf{sim}_{\mathsf{m}}$ first samples a random index $I \xleftarrow{\$} [n]$. Then it generates n key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \texttt{Gen}$, and once the adversary inputs $\diamond$ to the interfaces E emulating $\mathsf{1\text{-}AUT}_{n\to1}$, $\mathsf{sim}_{\mathsf{m}}$ outputs $\{(i, pk_i) \,|\, i \in [n]\}$ at the same interface. Whenever the adversary inputs $\diamond$ to the interfaces E emulating $\mathsf{A\text{-}INS}_{n\to1}$, $\mathsf{sim}_{\mathsf{m}}$ also inputs $\diamond$ to the interface E of $\mathsf{D\text{-}AUT}_{n\to1}$, obtaining a set $\mathfrak{D} \subseteq \{\underline{\texttt{cmt}}\} \times \mathbb{N} \times \mathcal{M}$.[5] It then outputs the set $\{(j, \underline{\texttt{cmt}}, m, \sigma) \,|\, (\sigma, \cdot) \leftarrow \texttt{Sgn}_{sk_I}(m), \exists\, (\underline{\texttt{cmt}}, j, m) \in \mathfrak{D}\}$ to E. Whenever the adversary inputs (m, σ) to the interface E emulating $\mathsf{A\text{-}INS}_{n\to1}$, $\mathsf{sim}_{\mathsf{m}}$ inputs $(\underline{\texttt{cmt}}, m)$ to the E interface of $\mathsf{D\text{-}AUT}_{n\to1}$.

The reduction system $\mathbf{C}_{\mathsf{m}}$ interacts with a system $[\mathbf{S}_1^-, \ldots, \mathbf{S}_n^-, \mathbf{K}]$, which is either $\mathbf{R}_{\mathsf{m}}$ or $\mathbf{S}_{\mathsf{m}}$. For any $j \in [t]$ such that $P_{\mathsf{msg}(j)}$, $\mathbf{C}_{\mathsf{m}}$ works by emulating $\pi_{\mathsf{PS}}[\mathsf{1\text{-}AUT}_{n\to1}, \mathsf{A\text{-}INS}_{n\to1}]$ during interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$, but replacing any call to $\texttt{Gen}$ by $\mathbf{K}$, any call to $\texttt{Sgn}_{sk_i}$ by $\mathbf{S}_i^-$, and using pk_i from $\mathbf{K}$ to implement $\texttt{Vrf}_{pk_i}$. Then clearly $\mathbf{C}_{\mathsf{m}}\,\mathbf{R}_{\mathsf{m}} \in (\pi_{\mathsf{PS}}[\mathsf{1\text{-}AUT}_{n\to1}, \mathsf{A\text{-}INS}_{n\to1}])^{[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]}$, and it is also easy to see that $\mathbf{C}_{\mathsf{m}}\,\mathbf{R}_{\mathsf{m}} \in (\mathsf{sim}_{\mathsf{m}}^{E}\,\mathsf{D\text{-}AUT}_{n\to1})^{[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]}$.

For any $j \in [t]$ such that $P_{\mathsf{hnd}(j)}$, the simulator $\mathsf{sim}_{\mathsf{h}}$ first sets $\mathcal{Q} \leftarrow \varnothing$. Then it generates n key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \texttt{Gen}$, and once the adversary inputs $\diamond$ to the interfaces E emulating $\mathsf{1\text{-}AUT}_{n\to1}$, $\mathsf{sim}_{\mathsf{h}}$ outputs $\{(i, pk_i) \,|\, i \in [n]\}$ at the same interface. Whenever the adversary inputs $\diamond$ to the interfaces E emulating $\mathsf{A\text{-}INS}_{n\to1}$, $\mathsf{sim}_{\mathsf{h}}$ also inputs $\diamond$ to the interface E of $\mathsf{D\text{-}AUT}_{n\to1}$, obtaining a set $\mathfrak{D} \subseteq \{\underline{\texttt{aut}}\} \times \mathbb{N}^2 \times \mathcal{M} \times [n]$.[6] It then outputs the set $\mathfrak{T} \doteq \{(j, \underline{\texttt{aut}}, m, \texttt{Sgn}_{sk_i}(m)) \,|\, \exists\, (\underline{\texttt{aut}}, j, j', m, i) \in \mathfrak{D}\}$ to E, and sets $\mathcal{Q} \leftarrow \mathcal{Q} \cup$

[5] Recall that $\mathsf{sim}_{\mathsf{m}}$ is working in an interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$ for $j \in [t]$ such that $P_{\mathsf{msg}(j)}$.

[6] Recall that $\mathsf{sim}_{\mathsf{h}}$ is working in an interval $[P_{\mathsf{fst}(j)}, P_{\mathsf{lst}(j)}]$ for $j \in [t]$ such that $P_{\mathsf{hnd}(j)}$.

$\{(j, m, \sigma, \tau) \,|\, (j, \underline{\mathtt{aut}}, m, \sigma, \tau) \in \mathfrak{T}\}$. Whenever the adversary inputs (m, σ, τ) to the interface E emulating $\textsf{A-INS}_{n\to1}$, if $(j, m, \sigma, \tau) \in \mathcal{Q}$ for some $j \in \mathbb{N}$, then $\textsf{sim}$ inputs j to the E interface of $\textsf{D-AUT}_{n\to1}$.

The reduction system $\mathbf{C}_{\textsf{h}}$ interacts with a system $[\langle \mathbf{S}_1, \mathbf{V}_1 \rangle, \ldots, \langle \mathbf{S}_n, \mathbf{V}_n \rangle, \mathbf{K}]$, which is either $\mathbf{R}_{\textsf{h}}$ or $\mathbf{S}_{\textsf{h}}$. For any $j \in [t]$ such that $P_{\textsf{hnd}(j)}$, $\mathbf{C}_{\textsf{h}}$ works by emulating $\pi_{\textsf{PS}}[\textsf{1-AUT}_{n\to1}, \textsf{A-INS}_{n\to1}]$ during interval $[P_{\textsf{fst}(j)}, P_{\textsf{lst}(j)}]$, but replacing any call to $\mathtt{Gen}$ by $\mathbf{K}$, any call to $\mathtt{Sgn}_{sk_i}$ by $\mathbf{S}_i$, and any call to $\mathtt{Vrf}_{pk_i}$ by $\mathbf{V}_i$. Then clearly $\mathbf{C}_{\textsf{h}}\, \mathbf{R}_{\textsf{h}} \in (\pi_{\textsf{PS}}[\textsf{1-AUT}_{n\to1}, \textsf{A-INS}_{n\to1}])^{[P_{\textsf{fst}(j)}, P_{\textsf{lst}(j)}]}$, and it is also easy to see that $\mathbf{C}_{\textsf{h}}\, \mathbf{R}_{\textsf{h}} \in (\textsf{sim}_{\textsf{h}}^{E}\ \textsf{D-AUT}_{n\to1})^{[P_{\textsf{fst}(j)}, P_{\textsf{lst}(j)}]}$. □

F.3 Receiver-Side Anonymous Authenticity

Theorem 3. *There exists a reduction system* $\mathbf{C}$ *such that, if a ring signature scheme* $\Sigma_{\textsf{RS}}$ *is* (n, ε)*-*$\textsf{UF-IK}$*-secure, then it is* $(n, \varepsilon^{\mathbf{C}})$*-composably secure.*

Proof. Let define systems

$$\mathbf{R} \doteq [\langle \mathbf{S}_{1,sk_1,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \ldots, \langle \mathbf{S}_{n,sk_n,\boldsymbol{pk}}, \mathbf{V}_{\boldsymbol{pk}} \rangle, \mathbf{K}_{\boldsymbol{pk}}],$$

$$\mathbf{S} \doteq [\underbrace{\langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle, \ldots, \langle \mathbf{S}_{I,sk_I,\boldsymbol{pk}}, \mathbf{V}^{\perp} \rangle}_{n \text{ times}}, \mathbf{K}_{\boldsymbol{pk}}],$$

for key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \mathtt{Gen}$, and random variable $I \xleftarrow{\$} [n]$. Then by Definition 11, $\Sigma_{\textsf{RS}}$ is such that $\mathbf{R} \approx_{\varepsilon} \mathbf{S}$. We now need to provide a simulator $\textsf{sim}$ and a reduction system $\mathbf{C}$ such that

$$\mathbf{CR} = \pi_{\textsf{RS}}[\textsf{1-AUT}_{n\circlearrowleft1}, \textsf{A-INS}_{n\to1}],$$

$$\mathbf{CS} = \textsf{sim}^{E}\ \textsf{RA-AUT}_{n\to1}.$$

This way we have $\Delta^{\mathbf{DC}}(\mathbf{R}, \mathbf{S}) \leq \varepsilon(\mathbf{DC}) = \varepsilon^{\mathbf{C}}(\mathbf{D})$, for any distinguisher $\mathbf{D}$, and therefore

$$\Delta^{\mathbf{D}}(\pi_{\textsf{BS}}[\textsf{1-AUT}_{n\circlearrowleft1}, \textsf{A-INS}_{n\to1}], \textsf{sim}^{E}\ \textsf{RA-AUT}_{n\to1}) \leq \varepsilon^{\mathbf{C}}(\mathbf{D}),$$

which by Lemma 6 proves the theorem.

The simulator $\textsf{sim}$ first sets $\mathcal{Q} \leftarrow \varnothing$ and samples a random index $I \xleftarrow{\$} [n]$. Then it generates n key-pairs $(sk_1, pk_1), \ldots, (sk_n, pk_n) \leftarrow \mathtt{Gen}$, sets $\boldsymbol{pk} \doteq (pk_1, \ldots, pk_n)$, and once the adversary inputs $\diamond$ to the interfaces E emulating $\textsf{1-AUT}_{n\circlearrowleft1}$, $\textsf{sim}$ outputs $\{(i, pk_i) \,|\, i \in [n]\}$ at the same interface. Whenever the adversary inputs $\diamond$ to the interfaces E emulating $\textsf{A-INS}_{n\to1}$, $\textsf{sim}$ also inputs $\diamond$ to the interface E of $\textsf{RA-AUT}_{n\to1}$, obtaining a set $\mathfrak{O} \subseteq \mathbb{N} \times \mathcal{M}$. It then outputs the set $\{(j, m, \mathtt{Sgn}_{sk_I,\boldsymbol{pk}}(m)) \,|\, \exists\, (j, m) \in \mathfrak{O}\}$ to E, and sets $\mathcal{Q} \leftarrow \mathcal{Q} \cup \mathfrak{O}$. Whenever the adversary inputs (m, σ) to the interface E emulating $\textsf{A-INS}_{n\to1}$, if $(j, m, \sigma) \in \mathcal{Q}$ for some $j \in \mathbb{N}$, then $\textsf{sim}$ inputs j to the E interface of $\textsf{RA-AUT}_{n\to1}$.

The reduction system $\mathbf{C}$ interacts with a system $[\langle \mathbf{S}_1, \mathbf{V}_1 \rangle, \ldots, \langle \mathbf{S}_n, \mathbf{V}_n \rangle, \mathbf{K}]$, which is either $\mathbf{R}$ or $\mathbf{S}$. $\mathbf{C}$ works by emulating $\pi_{\textsf{RS}}[\textsf{1-AUT}_{n\circlearrowleft1}, \textsf{A-INS}_{n\to1}]$, but replacing any call to $\mathtt{Gen}$ by $\mathbf{K}$, any call to $\mathtt{Sgn}_{i,sk_i,\boldsymbol{pk}}$ by $\mathbf{S}_i$, and any call to $\mathtt{Vrf}_{\boldsymbol{pk}}$ by $\mathbf{V}_i$ (for any $i \in [n]$). Then clearly $\mathbf{CR} = \pi_{\textsf{RS}}[\textsf{1-AUT}_{n\circlearrowleft1}, \textsf{A-INS}_{n\to1}]$, and it is also easy to see that $\mathbf{CS} = \textsf{sim}^{E}\ \textsf{RA-AUT}_{n\to1}$. □

G On Anonymous Signatures and Signcryption

In this section we briefly discuss anonymous signatures, the precursors of partial signatures. As we mentioned above, in the setting we are considering such scheme's security would not be possible to model, since we fixed the anonymous insecure channel $\mathsf{A\text{-}INS}_{n\to 1}$ as the assumed resource. But if we would strengthen this assumption, it would then be possible to model anonymous signatures' security as well. More concretely, if we additionally include to the assumed resources the anonymous confidential channel $\mathsf{A\text{-}CNF}_{n\to 1}$, as informally described in Section 1.2, it would then be possible to define composable security of a protocol π_{AS} using anonymous signatures as the construction of the anonymous secure channel $\mathsf{A\text{-}SEC}_{n\to 1}$, also informally described in Section 1.2, from $\mathsf{A\text{-}AUT}_{n\to 1}$, $\mathsf{A\text{-}INS}_{n\to 1}$, and $\mathsf{A\text{-}CNF}_{n\to 1}$, that is,

$$[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{A\text{-}CNF}_{n\to 1}] \xmapsto{\pi_{\mathsf{AS}}} \mathsf{A\text{-}SEC}_{n\to 1}.$$

Intuitively, π_{AS} would use $\mathsf{A\text{-}INS}_{n\to 1}$ to transmit the signature, and $\mathsf{A\text{-}CNF}_{n\to 1}$ for the message, so that the latter is not leaked to the adversary, which therefore cannot use it to verify and hence break anonymity.

Furthermore, the resource $\mathsf{A\text{-}CNF}_{n\to 1}$ could in principle be constructed from $\mathsf{1\text{-}AUT}_{n\leftarrow 1}$ and $\mathsf{A\text{-}INS}_{n\to 1}$ via a protocol π_{APKE} making use of a public-key encryption scheme satisfying appropriate anonymity properties. Then, similarly as the result from [KMO+13], one could show that

$$[\mathsf{1\text{-}AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}] \xmapsto{\pi_{\mathsf{APKE}}} \mathsf{A\text{-}CNF}_{n\to 1}.$$

Finally, one could compose the two schemes using the *encrypt-and-sign* paradigm, resulting in an anonymous signcryption scheme. The composed protocol $\pi_{\mathsf{SC}} = \pi_{\mathsf{AS}} \circ \pi_{\mathsf{APKE}}$ would then imply the construction

$$[\mathsf{1\text{-}AUT}_{n\to 1}, \mathsf{1\text{-}AUT}_{n\leftarrow 1}, \mathsf{A\text{-}INS}_{n\to 1}] \xmapsto{\pi_{\mathsf{SC}}} \mathsf{A\text{-}SEC}_{n\to 1}.$$