Research Article

Colleen M. Swanson and Douglas R. Stinson*

Unconditionally secure signature schemes revisited

DOI: 10.1515/jmc-2016-0002
Received January 4, 2016; revised March 14, 2016; accepted March 16, 2016

Abstract: Unconditionally secure signature (USS) schemes provide the ability to electronically sign documents without the reliance on computational assumptions needed in traditional digital signatures. Unlike digital signatures, USS schemes require that verification algorithms are not public – for any possible signer, a given user must have a different secret verification algorithm corresponding to that signer. Thus, any viable security definition for a USS scheme must carefully treat the subject of what constitutes a valid signature. That is, it is important to distinguish between signatures that are created using a user's signing algorithm and signatures that may satisfy one or more user verification algorithms. Moreover, given that each verifier has his own distinct verification algorithm, a USS scheme must necessarily handle the event of a disagreement. In this paper, we present a new security model for USS schemes that incorporates these notions, as well as give a formal treatment of dispute resolution and the trust assumptions required. We provide formal definitions of non-repudiation and transferability in the context of dispute resolution, and give sufficient conditions for a USS scheme to satisfy these properties. We then extend our basic framework to the setting of strong key-insulated signatures, which increase robustness against key exposure. Finally, we give security analyses for two constructions: Hanaoka et al.'s construction, which we show is secure in our basic USS model, and a key-insulated extension of this construction, which is secure in our strong key-insulated model.
This is an extended version of the conference paper [19], which appeared in ICITS 2011.

Keywords: Digital signatures, information-theoretic security, unconditionally secure signature schemes, key-insulation, dispute resolution

MSC 2010: 94A60

Communicated by: Spyros Magliveras

1 Introduction

Unconditionally secure signature (USS) schemes provide the ability to electronically sign documents without the reliance on computational assumptions needed in traditional digital signatures. That is, USS schemes are the analogue of digital signatures in the unconditionally secure cryptographic setting. The construction of such schemes is interesting not only from a theoretical perspective, but also from the viewpoint of ensuring security of information in the long term or designing schemes that are viable in a post-quantum world.

In traditional digital signatures, each user has a pair consisting of a secret signing algorithm and a public verification algorithm. Since user verification algorithms are public, anyone can verify whether a given signature was created by the claimed signer. Unlike digital signatures, USS schemes require that verification algorithms are not public – for any possible signer, each user must have a different secret verification algo-

Colleen M. Swanson: Department of Computer Science, University of California, Davis, One Shields Avenue, Davis, CA 95616, USA, e-mail: cmswan@ucdavis.edu
***Corresponding author: Douglas R. Stinson:** David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada, e-mail: dstinson@uwaterloo.ca

rithm corresponding to that signer. The consequence is that USS schemes necessarily have a limited number of users, and hence a limited number of entities with the ability to verify a given signature, each with their own special test. Thus, any viable security definition for a USS scheme must carefully treat the subject of what constitutes a valid signature. That is, it is important to distinguish between signatures that are created using a user's signing algorithm and signatures that may satisfy one or more user verification algorithms. Current research [7–9, 12, 16] has proposed various models for unconditionally secure signature schemes, but these models do not fully treat the implications of having multiple verification algorithms or analyze the need for (and trust questions associated with) having a dispute resolution mechanism. We address both of these issues in this paper.

Historically, there have been several attempts to create unconditionally secure constructions that satisfy security properties required for digital signatures, including non-repudiation, transferability, and unforgeability. Chaum and Roijakkers [2] introduced unconditionally secure signatures, proposing an interactive scheme that does not have transferability. Another approach to creating unconditionally secure signatures has been to enhance existing unconditionally secure message authentication codes (MACs), making these codes more robust in a signature setting. MACs clearly do not provide non-repudiation, as the sender and receiver compute authentication tags using the same algorithm. In addition, the need for a designated sender and receiver further limits the applicability of such schemes in a general signature setting.

Much research has been devoted to the removal of the standard MAC trust assumptions, in which both sender and receiver are assumed to be honest. In A^2-codes [10, 17, 18], the sender and receiver may be dishonest, but there is a trusted arbiter to resolve disputes; in A^3-codes [1, 4, 11], the arbiter is no longer trusted prior to dispute resolution, but is trusted to make an honest decision in event of a disagreement. Johansson [11] used A^3-codes to improve the construction of Chaum and Roijakkers by making it non-interactive, but the signatures produced by the scheme are not transferable, as the use of a designated receiver limits the verification of the signature to those who have the appropriate key. Multi-receiver authentication codes (MRAs) [3] and multi-receiver authentication codes with dynamic sender (DMRAs) [13] use a broadcast setting to relax the requirement for designation of receivers, and also, in the latter case, senders. These codes are not appropriate outside of a broadcast setting, however, as neither non-repudiation nor transferability are satisfied.

Unsurprisingly, the first security models for unconditionally secure signature schemes, including Johansson [11] and Hanaoka et al. [7, 8], drew upon the standard MAC security models. Shikata et al. [16] introduced a model using notions from public-key cryptography, which was also adopted in the work by Hara et al. [9] on blind signatures. Safavi-Naini et al. [12] presented a MAC-based model meant to encompass the notions developed by Shikata et al. In this work, we present a new security model. Our model is more general than the MAC-based models of Hanaoka et al. [7, 8] and Safavi-Naini et al. [12] and covers the attacks described in these works. Like that of Shikata et al. [16], our work is based on security notions from traditional public-key signature systems. However, our model differs from those in the existing literature in its careful treatment of the concept of a "valid" signature. Our aim is to provide a rigorous and natural security model that covers all reasonable attacks.

In addition, we analyze a construction of Hanaoka et al. [7] in our model and provide a proof of security. We remark that while Hanaoka et al. make claims about the security of this construction in their model, they do not provide an analysis. In fact, security proofs are not provided for most of the constructions given in existing research. Thus, we feel it is useful to include our analysis of a basic unconditionally secure signature construction in our security model.

Our basic notion of security is easily extendable to a system with dispute resolution, which we argue is a necessary component of any USS scheme. Furthermore, our treatment of dispute resolution allows us to give formal definitions of non-repudiation and transferability. We show that a USS scheme that satisfies our unforgeability definition and has an appropriate dispute resolution method also satisfies non-repudiation and transferability, both of which are required properties for any reasonable signature scheme. Finally, we define various dispute resolution methods and examine the amount of trust each requires.

An advantage of our security framework for USS schemes is its flexibility; standard security properties from the literature, such as *strong key insulation* [5, 6], can be incorporated into our basic model in a nat-

ural way. In key-insulated signature schemes, constructions are designed to be robust against signing-key exposure; this is done by splitting a user's signing information between a physically secure device (which stores the user's master key) and an insecure device (which is responsible for actually signing messages using temporary signing keys). We explore the notion of unconditionally secure strong key-insulated signatures in Sections 8 and 9, drawing from the work of Seito et al. [14] and Seito and Shikata [15] on unconditionally secure key-insulated multi-receiver authentication codes and key agreement. In particular, we give a formal extension of our security model to the strong key-insulation setting and present a construction that is secure in a restricted version of our model.

An outline of our paper is as follows. In Section 2, we give a basic definition of a USS scheme, before moving to an informal treatment of the desired security properties. We then define a formal security model in Section 3. We introduce the notion of dispute resolution and give examples of possible dispute resolution methods in Section 4; we then formally define dispute resolution in Section 4 and explore the impact of dispute resolution on our basic security notions of unforgeability, non-repudiation, and transferability. In Section 6, we compare our work with that of previous literature. We analyze the construction of Hanaoka et al. [7] in Section 7. In Sections 8 and 9, we give a formal treatment of USS schemes with strong key-insulation and then we present our construction. Finally, we give some concluding remarks in Section 10.

2 Preliminaries

We require the following definitions.

Definition 2.1. An unconditionally secure signature scheme (or USS scheme) Π consists of a tuple $(\mathcal{U}, \mathcal{X}, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ satisfying the following:

- The set $\mathcal{U} = \{U_1, \ldots, U_n\}$ consists of n possible users, $\mathcal{X}$ is a finite set of possible messages, and Σ is a finite set of possible signatures.
- The *key-generation algorithm* Gen takes as input 1^k, where k is a security parameter, and outputs the signing algorithm Sign and the verification algorithm Vrfy.
- The *signing algorithm* $\mathsf{Sign}: \mathcal{X} \times \mathcal{U} \to \Sigma$ takes a message $x \in \mathcal{X}$ and a signer $U_\zeta \in \mathcal{U}$ as input, and outputs a signature $\sigma \in \Sigma$. For each $U_\zeta \in \mathcal{U}$, we let Sign_ζ denote the algorithm $\mathsf{Sign}(\cdot, U_\zeta)$.
- The *verification algorithm* $\mathsf{Vrfy}: \mathcal{X} \times \Sigma \times \mathcal{U} \times \mathcal{U} \to \{True, False\}$ takes as input a message $x \in \mathcal{X}$, a signature $\sigma \in \Sigma$, a signer $U_\zeta \in \mathcal{U}$, and a verifier $U_\nu \in \mathcal{U}$, and outputs either *True* or *False*. For each user U_ν, we let Vrfy_ν denote the algorithm $\mathsf{Vrfy}(\cdot, \cdot, \cdot, U_\nu)$.

It is required that, for every k, for every pair $(\mathsf{Sign}, \mathsf{Vrfy})$ output by $\mathsf{Gen}(1^k)$, for every pair $U_\zeta, U_\nu \in \mathcal{U}$, and for every $x \in \mathcal{X}$, it holds that

$$\mathsf{Vrfy}_\nu(x, \mathsf{Sign}_\zeta(x), U_\zeta) = True.$$

Remark 2.2. We are treating *deterministic* signature schemes only, in the sense that Sign and Vrfy are deterministic, although the above definition can easily be extended to the randomized setting. In practice, we typically also want Sign and Vrfy to be polynomial-time algorithms for efficiency. The point of USS schemes is to guarantee security against powerful adversaries, even those who are computationally unlimited.

We now define the concepts of authentic, acceptable, and fraudulent signatures. Distinguishing these three concepts is one of the main themes of this section.

Definition 2.3. A signature $\sigma \in \Sigma$ on a message $x \in \mathcal{X}$ is *ζ-authentic* if $\sigma = \mathsf{Sign}_\zeta(x)$.

Definition 2.4. A signature $\sigma \in \Sigma$ on a message $x \in \mathcal{X}$ is *(ζ, ν)-acceptable* if $\mathsf{Vrfy}_\nu(x, \sigma, U_\zeta) = True$.

Definition 2.5. A signature $\sigma \in \Sigma$ on a message $x \in \mathcal{X}$ is *(ζ, ν)-fraudulent* if σ is (ζ, ν)-acceptable but not ζ-authentic.

Remark 2.6. In practice, we assume the existence of a trusted initializer TI who takes responsibility for scheme set up and key distribution. That is, the TI runs $\mathsf{Gen}(1^k)$ and securely distributes signing and verifi-

cation keys to the appropriate users. Participants cannot create their own signing information and distribute corresponding verification keys to the other users, as in this case each user U_ζ would be able to create a (ζ, ν)-fraudulent signature for all $U_\nu \in \mathcal{U}$. While it might be possible to avoid this problem by using a "group computation" approach to create and distribute the necessary scheme information, for simplicity we assume the existence of a TI.

2.1 Security notions

Informally, a secure signature scheme should satisfy the following three properties:

1. *Unforgeability*: Except with negligible probability with respect to the given security parameter k, it should not be possible to create a "valid" signature without the corresponding signing algorithm.
2. *Non-repudiation*: Except with negligible probability with respect to the given security parameter k, a signer should be unable to repudiate a legitimate signature that he has created.
3. *Transferability*: Except with negligible probability with respect to the given security parameter k, if a verifier accepts a signature, he can be confident that any other verifier will also accept it.

One objective of this paper is to formalize these notions in the unconditionally secure setting; we provide precise definitions in Sections 3 and 4. In contrast to the usual public-key setting, the requirements of non-repudiation and transferability are not guaranteed in a USS scheme that satisfies the above intuitive notion of unforgeability. For "ordinary" digital signatures, non-repudiation is a consequence of unforgeability: a signature is considered "valid" if it passes a verification test, and it should be infeasible for anyone to create such a signature without knowledge of the secret signing algorithm. Thus, assuming the signing algorithm is not known to some third party, the signer cannot create a signature and later repudiate it. Transferability of digital signatures is guaranteed since there is a single, public verification algorithm.

In USS schemes, the concept of a "valid" signature requires clarification. Given sufficient computation time, a verifier is always capable of finding a signature that passes his own, secret verification test, so we cannot define the validity of a signature based on whether it passes a given user's verification algorithm. Indeed, there must be signatures that pass a given user's verification algorithm but that could not have been created with the signer's signing algorithm; otherwise the scheme does not satisfy unforgeability. Similarly, each verifier's verification algorithm must be different, or a given verifier may be able to present a signature acceptable to any verifier who possesses the same verification algorithm. A "valid" signature, then, must be created using the signer's signing algorithm, and it should be infeasible for anyone to create a signature that *appears* valid to other, non-colluding users, or the scheme does not have the properties of unforgeability, non-repudiation, and transferability. In particular, we have the following observations.

Theorem 2.7. *A necessary condition for a USS scheme to satisfy unforgeability is the existence of* (ζ, ν)*-fraudulent signatures for* $\zeta \neq \nu$.

Proof. Given sufficient computation time, a verifier U_ν can use his verification algorithm to create a (ζ, ν)-acceptable signature for any $\zeta \neq \nu$. If there are no (ζ, ν)-fraudulent signatures, then all signatures produced in this fashion must be ζ-authentic, and therefore they are successful forgeries. □

Theorem 2.8. *A USS scheme that satisfies unforgeability has the property that* $\mathsf{Vrfy}_\nu(\cdot,\cdot,\cdot) \neq \mathsf{Vrfy}_\ell(\cdot,\cdot,\cdot)$ *for* $\nu \neq \ell$.

Proof. Suppose that $\mathsf{Vrfy}_\nu(\cdot,\cdot,\cdot) = \mathsf{Vrfy}_\ell(\cdot,\cdot,\cdot)$ where $\nu \neq \ell$. Given sufficient computation time, U_ν can create a (ζ, ν)-acceptable signed message, (x, σ). Because $\mathsf{Vrfy}_\nu(\cdot,\cdot,\cdot) = \mathsf{Vrfy}_\ell(\cdot,\cdot,\cdot)$, it follows immediately that (x, σ) is (ζ, ℓ)-acceptable. This implies that the user U_ℓ will accept (x, σ) as a valid signature, but (x, σ) was not created by U_ζ. □

3 Formal security model

We now develop a formal security model for USS schemes. Our security definition is comparable to the notion of signatures secure against existential forgery under adaptive chosen message attacks in the case of public-key signature schemes. However, our definition takes into account the distinctive characteristics of the unconditional security setting, in particular the existence (and necessity) of fraudulent signatures and multiple verification algorithms.

We specify two types of existential forgery. In our setting, an "existential" forgery is either a (ζ, ν)-fraudulent signature created without the help of the verifier U_ν, or a ζ-authentic signature created without the help of the signer U_ζ. If a USS scheme is secure, then both of these types of forgeries should be infeasible for an adversary to create.

We need the following oracles for our security definition:

- The $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ *oracle*; this oracle takes as input a message x and outputs an ℓ-authentic signature for the message x.
- The $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot,\cdot,\cdot)$ *oracle*; this oracle takes as input a signature pair (x, σ) and a signer U_ζ, and runs user U_ℓ's verification algorithm on input (x, σ, U_ζ), outputting *True* or *False*.

Definition 3.1. Let $\Pi = (\mathcal{U}, X, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k, let the set $C \subseteq \mathcal{U}$ be a coalition of at most ω users, and let ψ_S and ψ_V be positive integers. We define the following *signature game* $\text{Sig-forge}_{C,\Pi}(k)$ with target signer U_ζ and verifier U_ν:

1. $\mathsf{Gen}(1^k)$ is run to obtain the pair $(\mathsf{Sign}, \mathsf{Vrfy})$.
2. The coalition C is given bounded access to the $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ and $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot,\cdot,U_\zeta)$ oracles for ℓ satisfying $U_\ell \notin C$. In particular, C is allowed a total of ψ_S and ψ_V queries to the $\mathsf{Sign}^{\mathcal{O}}$ and $\mathsf{Vrfy}^{\mathcal{O}}$ oracles, respectively, with at most $\psi_S/(n - |C|)$ queries to $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ for each ℓ satisfying $U_\ell \notin C$. It should be noted that C has unlimited access to the signing and verification algorithms of any $U_\ell \in C$. We let $\mathcal{Q}$ denote the set of messages that the coalition submitted as queries to the oracle $\mathsf{Sign}_\zeta^{\mathcal{O}}(\cdot)$. Note that $\mathcal{Q}$ does not contain messages submitted as queries to $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ for $\ell \neq \zeta$.
3. The coalition C outputs a signature pair (x, σ).
4. The output of the game is defined to be 1 if and only if one of the following conditions is met:
   a. $U_\nu \notin C$ and σ is a (ζ, ν)-fraudulent signature on x; or
   b. $U_\zeta \notin C$, $x \notin \mathcal{Q}$, and σ is a ζ-authentic signature on x.

Definition 3.2. Let $\Pi = (\mathcal{U}, X, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k and let $\epsilon(k)$ be a negligible function of k. We say Π is $(\omega, \psi_S, \psi_V, \epsilon)$-*unforgeable* if for all coalitions C of at most ω possibly colluding users, and all choices of target signer U_ζ and verifier U_ν, it holds that

$$\Pr[\text{Sig-forge}_{C,\Pi}(k) = 1] \leq \epsilon(k).$$

Remark 3.3. Another option is to include a $\mathsf{Fraud}_{(\zeta,\nu)}^{\mathcal{O}}(\cdot)$ *oracle*; this oracle takes as input a message x and outputs a (ζ, ν)-fraudulent signature on x. Providing certain (ζ, ν)-fraudulent signatures to the adversary could only increase his chances of ultimately constructing a new (ζ, ν)-fraudulent signature. Thus this would constitute a stronger security model than the one we consider. On the other hand, it is hard to envisage a practical scenario where an adversary would have this kind of additional information about a verifier whom the adversary is attempting to deceive. Therefore we do not include the $\mathsf{Fraud}^{\mathcal{O}}$ oracle in our basic model of USS schemes. However, it would be straightforward to modify our model to include these oracles, if desired.

We observe that a scheme meeting the unforgeability requirement of Definition 3.2 satisfies our intuitive notions of non-repudiation and transferability. We explain these relationships in the following observations, noting that formal definitions of non-repudiation and transferability are intrinsically linked to the dispute resolution process, and so are provided later, in Section 4. We will formalize these observations in Theorems 5.8 and 5.12.

Observation 3.4. *An* $(\omega, \psi_S, \psi_V, \epsilon)$*-unforgeable USS scheme* Π *provides non-repudiation.*

Proof. Suppose that Π is $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable. Then U_ζ cannot repudiate a given ζ-authentic signature σ, as Definition 3.2 guarantees that σ can be created without U_ζ only with negligible probability (as Condition 4b of Definition 3.1 holds only with negligible probability). Thus U_ζ cannot claim that other users may have created σ. The other possibility for a signer U_ζ to repudiate a signature on a message given to U_v is if the signature is (ζ, v)-fraudulent. Definition 3.2 also implies that U_ζ cannot create a (ζ, v)-fraudulent signature (even with the help of $\omega - 1$ other users not including U_v) except with negligible probability, as Condition 4a of Definition 3.1 is assumed to not hold (except with negligible probability). □

Observation 3.5. *An* $(\omega, \psi_S, \psi_V, \epsilon)$*-unforgeable USS scheme* Π *provides transferability.*

Proof. In order for a signature σ to be non-transferable from U_v to U_ℓ, the signature σ must be (ζ, v)-acceptable, but not (ζ, ℓ)-acceptable, where $v \neq \ell$. If σ were ζ-authentic, it would also be (ζ, ℓ)-acceptable. Therefore σ must be (ζ, v)-fraudulent. However, Definition 3.2 implies a (ζ, v)-fraudulent signature cannot be created without the assistance of U_v, except with negligible probability. □

From the point of view of a verifier, a scheme meeting Definition 3.2 gives reasonable assurance of the validity of a received signature. If a verifier U_v receives a signature pair (x, σ) purportedly from U_ζ, then U_v accepts the signature so long as σ is (ζ, v)-acceptable for the message x. In this case, there are only two possibilities: either σ is ζ-authentic or (ζ, v)-fraudulent for the message x. If σ is ζ-authentic, then a coalition that does not include the signer U_ζ has only a negligible probability of creating σ by Condition 4b of Definition 3.1. If σ is (ζ, v)-fraudulent, then Condition 4a of Definition 3.1 guarantees that a coalition that does not include U_v cannot create σ, except with negligible probability.

4 Dispute resolution

Given that each verifier has his own distinct verification algorithm, a USS scheme must necessarily handle the event of a disagreement. That is, since there is no public verification method as in traditional digital signatures, a USS scheme must have a mechanism to determine the authenticity of a signature when some subset of users disagree whether a given signature should be accepted. In particular, dispute resolution is necessary to convince an outsider of the authenticity of a disputed signature. In traditional digital signatures, there are no outsiders to the scheme, in the sense that everyone has access to the public verification method. In our setting, however, the number of participants (and therefore their access to verification algorithms) is limited. Dispute resolution is a method that effectively deals with the need for resolution of disagreements in, for example, a court setting. Typically, dispute resolution involves all the users voting on the validity of a signature, or alternatively, a trusted arbiter stating whether a signature is valid.

The manner in which a dispute resolution mechanism may be invoked necessarily affects the security of the overall scheme. In particular, we should not allow users to invoke the dispute resolution mechanism an arbitrary number of times. If users have unlimited access, it may be possible for a coalition to use dispute resolution as a type of verification oracle against a target signer U_ζ. As this is undesirable, we need to limit access to dispute resolution in a reasonable way. One simple possibility is to limit dispute resolution to once per scheme. That is, once dispute resolution has been invoked, we require that the users request the TI to generate new signing and verification keys. This may be reasonable because dispute resolution necessarily implies the existence of a lying (or otherwise compromised) user, and we find it unlikely that users will want to continue the current scheme with the dishonest (or compromised) user in question; we discuss these concepts in more detail in Remark 4.5. That said, it may be desirable to include a mechanism by which to determine and punish cheaters – such a mechanism may be useful if multiple calls to the dispute resolution are desired, or to determine which users should not be included in a scheme reset.

We focus on the case in which dispute resolution causes a scheme reset. We begin with some basic concepts and then provide and analyze examples of possible dispute resolution mechanisms. Ideally, the dispute

resolution process validates a signature if and only if the signature is authentic, i.e., the signature was produced by the purported signer. This leads to the following definitions.

Definition 4.1. A *dispute resolution method* $\mathcal{DR}$ for a USS scheme Π is a procedure invoked when a pair of users U_ℓ, $U_{\ell'} \in \mathcal{U}$ disagrees as to the validity of a given signature (x, σ), purportedly signed by U_ζ. Here U_ℓ (respectively, $U_{\ell'}$) may be any user in $\mathcal{U}$, including U_ζ. The procedure $\mathcal{DR}$ consists of an algorithm DR that takes as input a signature pair (x, σ) and a purported signer U_ζ, and outputs a value in $\{Valid, Invalid\}$, subject to the following rules:

1. If DR outputs *Valid*, then (x, σ) must subsequently be accepted as a ζ-authentic signature on x by all users.
2. If DR outputs *Invalid*, then (x, σ) must subsequently be rejected by all users.

We remark that the algorithm DR may have access to additional (secret) scheme information, as specified by the particular dispute resolution method.

The following definitions capture the desirable properties of a given $\mathcal{DR}$.

Definition 4.2 (Soundness). Let Π be a USS scheme and let $\mathcal{DR}$ be a dispute resolution method for Π. We say $\mathcal{DR}$ is *sound* if, whenever σ is not a ζ-authentic signature on x, then $\mathsf{DR}((x, \sigma), U_\zeta)$ outputs *Invalid*.

Definition 4.3 (Completeness). Let Π be a USS scheme and let $\mathcal{DR}$ be a dispute resolution method for Π. We say $\mathcal{DR}$ is *complete* if, whenever σ is a ζ-authentic signature on x, then $\mathsf{DR}((x, \sigma), U_\zeta)$ outputs *Valid*.

Definition 4.4 (Correctness). Let Π be a USS scheme and let $\mathcal{DR}$ be a dispute resolution method for Π. If $\mathcal{DR}$ is both sound and complete, we say $\mathcal{DR}$ is *correct*.

Remark 4.5. A correct dispute resolution method $\mathcal{DR}$ is useful in terms of identifying and punishing users who are cheating (or alternatively whose secret information has been compromised). To see this, suppose that a signature σ with purported signer U_ζ is given to $\mathcal{DR}$ by two users U_ℓ and $U_{\ell'}$. Without loss of generality, suppose U_ℓ claims σ should be accepted and $U_{\ell'}$ claims σ should be rejected. Then if the output of $\mathcal{DR}$ is *Valid*, soundness implies that σ is ζ-authentic. In this case, the user $U_{\ell'}$ is either dishonest or otherwise compromised. If, on the other hand, the output of $\mathcal{DR}$ is *Invalid*, completeness implies that σ is not ζ-authentic. In this case, the user U_ℓ is either dishonest or otherwise compromised. Here, by *otherwise compromised*, we are recognizing the possibility that a user's secret information may become *unintentionally* known to an adversary (i.e., a coalition of dishonest users), but the user in question is honest. This might happen, for example, due to insecure storage of signing and/or verification keys.

We define three dispute resolution methods and examine the level of honesty required in each scheme. In particular, we wish to define trust assumptions sufficient to ensure the correctness of these dispute resolution methods. That is, we consider the degree of trust a group of users should have in order to use a particular dispute resolution method.

Definition 4.6. We have the following dispute resolution methods, assuming a disputed signature σ on message x with purported signer U_ζ:

- *Omniscient Arbiter (OA) Dispute Resolution*: Designate an arbiter equipped with all of the USS scheme setup information. The signature σ is considered valid if the arbiter, using his knowledge of all the signing and verification algorithms, accepts the signature as authentic. Here we assume the arbiter is honest.
- *Verifier-Equivalent Arbiter (VEA) Dispute Resolution*: Designate an arbiter equipped with his own verification algorithm, $\mathsf{Vrfy}_{\mathcal{A}}$, (i.e., the arbiter is a *glorified verifier*). The arbiter tests the authenticity of the signature σ by running $\mathsf{Vrfy}_{\mathcal{A}}(x, \sigma, U_\zeta)$; the signature is considered valid if $\mathsf{Vrfy}_{\mathcal{A}}(x, \sigma, U_\zeta)$ outputs *True*. Here we assume the arbiter is honest. We remark that the arbiter may or may not be a normal user in the scheme, although assuming the arbiter is honest may be more reasonable if the arbiter is not otherwise involved with the scheme.
- *Majority Vote (MV) Dispute Resolution*: Here we resolve disputes by having the users vote on the validity of the signature σ. Each user is responsible for running his verification algorithm on (x, σ, U_ζ) and casting a *valid vote* if his verification algorithm outputs *True* and an *invalid vote* otherwise. The signature is

considered valid if a prespecified threshold of *valid* votes are cast; here we consider the case of a majority threshold and assume all users vote. We assume that a majority of users are honest.

In the case of OA dispute resolution, it is clear that we require the arbiter to be honest, as he has all the necessary information to sign and verify documents on behalf of other users. That is, a USS scheme Π with OA dispute resolution clearly cannot satisfy any unforgeability condition unless the arbiter is honest, as the arbiter has all the necessary information to sign messages on behalf of users. Moreover, provided that the arbiter is honest, this dispute resolution method is both sound and complete, as the arbiter is able to determine the authenticity of a given signature and behave appropriately. In fact, the correctness of OA dispute resolution with an honest arbiter is independent of the security of the underlying scheme. Correctness implies the arbiter's ability to identify signatures that are ζ-authentic for the purported signer U_ζ, which is independent from the problem of preventing other users from creating a ζ-authentic signature without U_ζ's help. However, it is of course still the case that correct dispute resolution is only useful in conjunction with an unforgeable USS scheme. To summarize, we have the following result:

Theorem 4.7. *Let Π be a USS scheme and let $\mathcal{DR}$ be an OA dispute resolution method for Π with an honest arbiter. Then $\mathcal{DR}$ is correct.*

Remark 4.8. Although we focus on deterministic signature schemes in this section, an interesting observation with respect to OA dispute resolution arises in the case of randomized signature schemes. In particular, if the signature scheme is randomized, then the arbiter may have to be computationally unbounded in order to perform dispute resolution. That is, if a purported signer claims a disputed signature is not valid, the arbiter may have to search an exponential space. This issue does not arise if the purported signer does not dispute the validity of the signature, however, as in this case the signer can simply reveal the randomness used to produce the disputed signature.

In the next two theorems, we present trust assumptions sufficient to achieve correctness in VEA and MV dispute resolution. For these methods, it is necessary to consider the security properties of the underlying signature scheme Π.

Theorem 4.9. *Let Π be an $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable USS scheme and let $\mathcal{DR}$ be a VEA dispute resolution method for Π with an honest arbiter. Then $\mathcal{DR}$ is correct in the presence of a coalition of users of maximum size ω, except with negligible probability.*

Proof. Suppose we have a disputed signature σ on message x with purported signer U_ζ. The arbiter $\mathcal{A}$ outputs *Valid* if and only if σ is $(\zeta, \mathcal{A})$-acceptable.

Given that Π satisfies our unforgeability definition, a coalition of maximum size ω cannot produce a signature that is $(\zeta, \mathcal{A})$-fraudulent without $\mathcal{A}$'s help, except with negligible probability. That is, an honest arbiter $\mathcal{A}$ outputs *Valid* exactly when σ is ζ-authentic (except with negligible probability). □

Theorem 4.10. *Let Π be an $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable USS scheme and let $\mathcal{DR}$ be an MV dispute resolution method for Π. Then $\mathcal{DR}$ is correct in the presence of a coalition of dishonest users of maximum size $\min\{\omega, \lfloor \frac{n-1}{2} \rfloor\}$, except with negligible probability.*

Proof. Suppose we have a disputed signature σ on message x with purported signer U_ζ. Consider a coalition C of size at most $\min\{\omega, \lfloor \frac{n-1}{2} \rfloor\}$. If x is ζ-authentic, then any honest $U_\ell \notin C$ will cast a *Valid* vote. The coalition C can attempt to ensure that x is rejected by having each member cast an *Invalid* vote, but as long as a majority of users are honest, x will be accepted by the dispute resolution process. If x is not ζ-authentic, then since Π is $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable, we have that x is not (ζ, ℓ)-fraudulent for any honest $U_\ell \notin C$ except with negligible probability. That is, any honest U_ℓ will (with overwhelming probability) cast an *Invalid* vote. The members of C can attempt to have x accepted by having each member cast a *Valid* vote, but given that a majority of users are honest, this approach works with only negligible probability. □

Remark 4.11. The proof of Theorem 4.9 establishes that the correctness of the VEA dispute resolution method depends on how easy it is to construct signatures which are $(\zeta, \mathcal{A})$-acceptable for users $U_\zeta \in \mathcal{U}$ in

the underlying scheme Π. In particular, it is easy to see that if Π does not satisfy Definition 3.2 for some ω with respect to output 4a of the security game Sig-forge$_{C,\Pi}(k)$ (as defined in Definition 3.1), then the VEA method fails to be correct, even with an honest arbiter. In addition, if we consider the maximum ω for which Π satisfies the above unforgeability criterion, it is easy to see that the VEA method fails to be correct in the presence of more than ω colluding users. Similar observations hold for Theorem 4.10 with respect to the MV dispute resolution method.

As observed above, we achieve correctness of the VEA method by assuming that the arbiter is honest. Achieving soundness and completeness is not as clear if we weaken this honesty requirement, however. In the typical VEA dispute resolution methods considered in current literature [9, 12, 16], the arbiter is assumed to be a glorified verifier, with the same type of keying information as an arbitrary verifier. The arbiter is assumed to follow the rules of the dispute resolution method honestly and is otherwise treated as a normal user in the context of the security model, i.e., he is allowed to be dishonest otherwise. That is, the arbiter is allowed to be a member of the coalition attempting to create a forgery, but he is expected to follow the dispute resolution process itself honestly. We refer to this set of trust assumptions as the *split trust assumption*. We argue that the split trust assumption is problematic, however, and should likely be abandoned. In particular, if we consider VEA dispute resolution where we allow the arbiter to be part of a given coalition, then soundness is no longer guaranteed.

The arbiter's distinct role in the dispute resolution method necessitates a more careful study of the arbiter, and therefore treating the arbiter as a normal verifier in the context of the security model is insufficient. While it is obvious an arbiter who is dishonest during dispute resolution can cause a fraudulent signature to be deemed valid, we cannot allow the arbiter to be dishonest before dispute resolution either, contrary to the claims of Safavi-Naini et al. [12] and Shikata et al. [16]. In particular, the VEA dispute resolution method does not achieve soundness under the split trust assumption due to the existence of a new type of forgery introduced by the dispute resolution process, which we term a *dispute-enabled forgery*:

Definition 4.12. Let Π be a USS scheme and let $\mathcal{DR}$ be a dispute resolution method for Π. We say a signature σ on a message $x \in \mathcal{X}$ is a *dispute-enabled forgery for signer* U_ζ if σ is not ζ-authentic, but $\mathsf{DR}((x, \sigma), U_\zeta)$ outputs *Valid*.

In fact, the proof of Theorem 4.9 indicates why the split trust assumption is problematic: an honest arbiter $\mathcal{A}$ outputs *Valid* during dispute resolution if and only if the signature is $(\zeta, \mathcal{A})$-acceptable for purported signer U_ζ. But if we allow $\mathcal{A}$ to be dishonest prior to dispute resolution, then $\mathcal{A}$ can produce a signature x that is $(\zeta, \mathcal{A})$-fraudulent. In this case, $\mathcal{A}$'s verification algorithm outputs *True* on input x with signer U_ζ, so x is a dispute-enabled forgery. We remark that the case of MV may be viewed as a generalized version of VEA dispute resolution and the security concerns are similar.

The main observation is that a cheating arbiter $\mathcal{A}$ (or, in the case of MV dispute resolution, a collusion of a majority of verifiers) can successfully forge a (ζ, v)-fraudulent signature for any cooperating user U_v. Hence, VEA and MV dispute resolution do not protect the signer against a dishonest arbiter (or a dishonest majority of verifiers) *under the split trust assumption*, since dispute-enabled forgeries exist. From the perspective of signer security, the split trust assumption is certainly not reasonable.

From the perspective of verifier security, it is interesting to note that both the VEA and MV methods are acceptable under the split trust assumption. This is a consequence of the fact that both the VEA and MV methods are complete provided that the dispute resolution process itself is performed honestly. We show in Theorems 5.8 and 5.12 that completeness is sufficient for an $(\omega, \psi_S, \psi_V, \epsilon)$-USS scheme Π with dispute resolution $\mathcal{DR}$ to provide non-repudiation and transferability. That is, the VEA and MV methods do not require the arbiter(s) to be honest prior to dispute resolution in order to achieve non-repudiation and transferability. As seen above, however, the VEA and MV methods require the arbiter(s) to be honest prior to dispute resolution in order to achieve soundness. In this sense, we see that VEA and MV dispute resolution under the split trust assumption provide similar *verifier security* to OA dispute resolution with an honest arbiter (in that non-repudiation and transferability are assured), but they fail to provide similar *signer security* (in that unforgeability is not assured).

Nonetheless, we argue that a more reasonable approach to dispute resolution is to assume the possibility of cheating both *before and during* dispute resolution. In this case, we see that for the VEA method, we must have an honest arbiter $\mathcal{A}$, and for the MV method, we require that a majority of users are honest.

With these examples in mind, we give a formal treatment of dispute resolution in the following section.

5 A formal treatment of dispute resolution

The possibility of dispute-enabled forgeries requires an extension to the unforgeability requirement of a USS scheme. Although unforgeability (unlike transferability and non-repudiation) is not intrinsically linked to the dispute resolution process, we need to ensure that the dispute resolution process itself does not weaken the overall security of the scheme.

Definition 5.1. Let Π be a USS scheme and let $\mathcal{DR}$ be a dispute resolution method for Π. We extend the signature game Sig-forge$_{C,\Pi}(k)$ to the *signature game* $\mathcal{DR}$-Sig-forge$_{C,\Pi}(k)$ by adjusting Definition 3.1 as follows.

We make the following changes to Step 4:

4. We add the following to the list of possible conditions for which the output of the game is 1:
   c. $U_\zeta \notin C$, σ is not ζ-authentic, but $\mathsf{DR}((x, \sigma), U_\zeta)$ outputs *Valid*.

Definition 5.2. Let $\Pi = (\mathcal{U}, \mathcal{X}, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k and let $\mathcal{DR}$ be a dispute resolution method for Π. Let $\epsilon(k)$ be a negligible function of k. We say the pair $(\Pi, \mathcal{DR})$ is $\mathcal{DR}$-*unforgeable* with parameters $(\omega, \psi_S, \psi_V, \epsilon)$ if for all coalitions C of at most ω possibly colluding users, and all choices of target signer U_ζ and verifier U_v, it holds that

$$\Pr[\mathcal{DR}\text{-Sig-forge}_{C,\Pi}(k) = 1] \leq \epsilon(k).$$

Remark 5.3. Here we model an attack in which a call to dispute resolution necessitates an immediate scheme reset. If we wish to account for the possibility of multiple calls to the dispute resolution method, we can allow C bounded access to a new oracle, the $\mathcal{DR}(\cdot, \cdot, \cdot)$ *oracle*, which takes as input a signature pair (x, σ) and a signer U_ζ and simulates the dispute resolution method $\mathcal{DR}$ on input (x, σ, U_ζ), outputting either *Valid* or *Invalid*.

We now observe that given an underlying scheme Π satisfying our *original* definition of unforgeability (Definition 3.2), we can achieve our stronger definition of $\mathcal{DR}$-unforgeability by choosing a *sound* dispute resolution method $\mathcal{DR}$.

Theorem 5.4. *Let Π be an $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable USS scheme and let $\mathcal{DR}$ be a sound dispute resolution method for Π. Then the pair $(\Pi, \mathcal{DR})$ is* $\mathcal{DR}$-unforgeable *with parameters $(\omega, \psi_S, \psi_V, \epsilon)$.*

Proof. Since Π is $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable, we see that a coalition C of at most ω users cannot produce signatures satisfying Conditions 4a or 4b of Definition 5.1 except with negligible probability. If, in addition, the dispute resolution method $\mathcal{DR}$ is sound, then $\mathcal{DR}$ outputs *Invalid* when given a signature that is not ζ-authentic for (any choice of) target signer U_ζ, so C cannot produce a signature satisfying Condition 4c. □

Remark 5.5. Condition 4c of Definition 5.1 says that a possible successful output of $\mathcal{DR}$-Sig-forge$_{C,\Pi}(k)$ is a dispute-enabled forgery. Definition 5.2 implies that with high probability, the coalition C is unable to find a dispute-enabled forgery. In other words, the coalition C is able to produce a signature compromising the soundness of the dispute resolution method $\mathcal{DR}$ with only negligible probability.

We now discuss the properties of non-repudiation and transferability. As previously mentioned, both of these properties are intrinsically linked to dispute resolution. That is, the outcome of the chosen dispute resolution method determines the success or failure of these attacks. In particular, we show that completeness is sufficient to achieve both non-repudiation and transferability.

We remark that in order for the dispute resolution method to be invoked in the first place, there must be disagreement as to the validity of a given signature σ. In a *repudiation attack*, the signer U_ζ gives a (ζ, v)-

acceptable signature σ to the verifier U_v (i.e., σ appears valid to U_v) and then later denies the validity of σ. In this case, the signer U_ζ and the target verifier U_v will invoke the dispute resolution method. Similarly, for a *transferability attack*, a verifier U_v transfers a signature σ that is (ζ, v)-acceptable (i.e., σ appears valid to U_v) to another user U_ℓ, who rejects σ as invalid. Thus, the dispute resolution method is again invoked, this time by users U_v and U_ℓ. In this case, U_v is assumed to be honest, but we remark that it is also possible that U_ℓ is honest, in the sense that U_ℓ may genuinely believe the signature in question to be invalid. That said, it is also possible for U_ℓ to be part of the attempt to "trap" U_v (independently of whether or not the given signature is rejected by U_ℓ's verification algorithm). We now provide formal definitions of these two attacks.

Definition 5.6. Let $\Pi = (\mathcal{U}, \mathcal{X}, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k and let $\mathcal{DR}$ be a dispute resolution method for Π. Let the set $C \subseteq \mathcal{U}$ be a coalition of at most ω users, and let ψ_S and ψ_V be positive integers. We define the following *signature game* $\text{Repudiation}_{C,\Pi}(k)$ with signer $U_\zeta \in C$ and target verifier U_v satisfying $U_v \notin C$:

1. $\mathsf{Gen}(1^k)$ is run to obtain the pair $(\mathsf{Sign}, \mathsf{Vrfy})$.
2. The coalition C is given bounded access to the $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ and $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot, \cdot, U_\zeta)$ oracles for ℓ satisfying $U_\ell \notin C$. In particular, C is allowed a total of ψ_S and ψ_V queries to the $\mathsf{Sign}^{\mathcal{O}}$ and $\mathsf{Vrfy}^{\mathcal{O}}$ oracles, respectively, with at most $\psi_S/(n - |C|)$ queries to $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ for each ℓ satisfying $U_\ell \notin C$. It should be noted that C has unlimited access to the signing and verification algorithms of any $U_\ell \in C$.
3. The coalition C outputs a signature pair (x, σ).
4. The output of the game is defined to be 1 if and only if one of the following conditions is met:
   a. σ is (ζ, v)-fraudulent and the dispute resolution method $\mathcal{DR}$ (as invoked by U_ζ and U_v) rejects σ as *Invalid*.
   b. σ is ζ-authentic and the dispute resolution method $\mathcal{DR}$ (as invoked by U_ζ and U_v) rejects σ as *Invalid*.

Definition 5.7. Let $\Pi = (\mathcal{U}, \mathcal{X}, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k and let $\mathcal{DR}$ be a dispute resolution method for Π. Let $\epsilon(k)$ be a negligible function of k. We say the combined scheme $(\Pi, \mathcal{DR})$ satisfies *non-repudiation* with parameters $(\omega, \psi_S, \psi_V, \epsilon)$ if for all coalitions C of at most ω possibly colluding users, and for all choices of signer U_ζ and target verifier U_v, it holds that

$$\Pr[\text{Repudiation}_{C,\Pi}(k) = 1] \leq \epsilon(k).$$

In the following theorem, we demonstrate that a dispute resolution method $\mathcal{DR}$ that is complete, when combined with an underlying USS scheme Π that is unforgeable, suffices to ensure non-repudiation attacks are (highly) unlikely to succeed.

Theorem 5.8. *Let Π be an $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable USS scheme and let $\mathcal{DR}$ be a complete dispute resolution method for Π. Then $(\Pi, \mathcal{DR})$ provides non-repudiation.*

Proof. Assume Π does not provide non-repudiation, so with non-negligible probability, $\text{Repudiation}_{C,\Pi}(k)$ outputs 1. Suppose $\text{Repudiation}_{C,\Pi}(k)$ with signer U_ζ and target verifier U_v outputs 1. Then C has created a (ζ, v)-acceptable signature pair (x, σ), such that the dispute resolution method $\mathcal{DR}$ (as invoked by U_ζ and U_v) rejects σ as *Invalid*.

Now, σ is either ζ-authentic or (ζ, v)-fraudulent. If σ is ζ-authentic, then the dispute resolution method $\mathcal{DR}$ rejected a ζ-authentic signature and is not complete. Therefore, if $\mathcal{DR}$ is complete, then every such signature σ that yields an output of 1 in the game $\text{Repudiation}_{C,\Pi}(k)$ must be (ζ, v)-fraudulent. Thus, with non-negligible probability, C can create a (ζ, v)-fraudulent signature that satisfies Condition 4a of Definition 3.1, resulting in an output of 1 in the game $\text{Sig-forge}_{C,\Pi}(k)$ with target signer $U_\zeta \in C$ and verifier $U_v \notin C$. That is, Π is not $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable. □

Definition 5.9. Let $\Pi = (\mathcal{U}, \mathcal{X}, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k and let $\mathcal{DR}$ be a dispute resolution method for Π. Let the set $C \subseteq \mathcal{U}$ be a coalition of at most ω users, and let ψ_S and ψ_V be positive integers. We define the following *signature game* $\text{Non-transfer}_{C,\Pi}(k)$ with signer U_ζ and target verifier U_v, where $U_v \notin C$:

1. $\mathsf{Gen}(1^k)$ is run to obtain the pair $(\mathsf{Sign}, \mathsf{Vrfy})$.
2. The coalition C is given bounded access to the $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ and $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot, \cdot, U_\zeta)$ oracles for ℓ satisfying $U_\ell \notin C$. In particular, C is allowed a total of ψ_S and ψ_V queries to the $\mathsf{Sign}^{\mathcal{O}}$ and $\mathsf{Vrfy}^{\mathcal{O}}$ oracles, respectively, with at most $\psi_S/(n - |C|)$ queries to $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ for each ℓ satisfying $U_\ell \notin C$. It should be noted that C has unlimited access to the signing and verification algorithms of any $U_\ell \in C$.
3. The coalition C outputs a signature pair (x, σ).
4. The output of the game is defined to be 1 if and only if the following conditions are met:
   a. σ is (ζ, ν)-fraudulent and the dispute resolution method $\mathcal{DR}$, as invoked by U_ν and some user $U_\ell \in \mathcal{U}$, outputs *Invalid*.
   b. σ is ζ-authentic and the dispute resolution method $\mathcal{DR}$, as invoked by U_ν and some verifier $U_\ell \in C$, outputs *Invalid*.

Remark 5.10. The distinction between the two cases in part 4 of Definition 5.9 is with respect to the integrity of the users who invoke the dispute resolution method. In the first case, it is possible that an honest verifier $U_\ell \notin C$ for whom σ is not (ζ, ℓ)-fraudulent may be involved, hence (unwittingly) aiding the coalition in trapping the target verifier U_ν. If σ is ζ-authentic, then there is no such user, as all honest verifiers would accept σ, so a member of the coalition C must participate in invoking dispute resolution.

Definition 5.11. Let $\Pi = (\mathcal{U}, \mathcal{X}, \Sigma, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a USS scheme with security parameter k and let $\mathcal{DR}$ be a dispute resolution method for Π. Let $\epsilon(k)$ be a negligible function of k. We say the combined scheme $(\Pi, \mathcal{DR})$ satisfies *transferability* with parameters $(\omega, \psi_S, \psi_V, \epsilon)$ if for all choices of signer U_ζ and target verifier U_ν, it holds that

$$\Pr[\text{Non-transfer}_{C,\Pi}(k) = 1] \leq \epsilon(k).$$

The following theorem is similar to Theorem 5.8 and gives the corresponding result for transferability.

Theorem 5.12. *Let Π be an $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable USS scheme and let $\mathcal{DR}$ be a complete dispute resolution method for Π. Then $(\Pi, \mathcal{DR})$ satisfies transferability.*

Proof. Suppose Π does not provide transferability and further assume the game $\text{Non-transfer}_{C,\Pi}(k)$ outputs 1, with signer U_ζ and target verifier $U_\nu \notin C$. Then C output a signature pair (x, σ) such that σ is (ζ, ν)-acceptable and the dispute resolution method (as invoked by U_ν and some user U_ℓ) rejected σ as *Invalid*.

Now, σ is either ζ-authentic or (ζ, ν)-fraudulent. If σ is ζ-authentic, then the dispute resolution method $\mathcal{DR}$ rejected a ζ-authentic signature and is not complete. Therefore, if the $\mathcal{DR}$ is complete, then every such signature σ that yields an output of 1 in the game $\text{Non-transfer}_{C,\Pi}(k)$ must be (ζ, ν)-fraudulent. Thus, with non-negligible probability, C can create a (ζ, ν)-fraudulent signature that satisfies Condition 4a of Definition 3.1, resulting in an output of 1 in the game $\text{Sig-forge}_{C,\Pi}(k)$ with target signer $U_\zeta \in C$ and verifier $U_\nu \notin C$. That is, Π is not $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable. □

Together, Theorems 5.4, 5.8, and 5.12 provide sufficient conditions for a USS scheme Π and a dispute resolution method $\mathcal{DR}$ to satisfy the desired properties of unforgeability, non-repudiation, and transferability. In particular, it suffices to take Π to be $(\omega, \psi_S, \psi_V, \epsilon)$-unforgeable and $\mathcal{DR}$ to be sound and complete (i.e., correct). Furthermore, we remark that Condition 4b of Definition 5.6 and Condition 4b of Definition 5.9 both correspond to a demonstration of the lack of completeness of the associated $\mathcal{DR}$. That is, in a scheme that satisfies non-repudiation or transferability, it must be infeasible to find a signature pair that acts as a witness to the lack of completeness of the associated $\mathcal{DR}$.

6 Comparison with existing models

Our model differs from those in the existing literature in its careful treatment of ζ-authentic and (ζ, ν)-fraudulent signatures. In comparison to other works, our approach is most similar to that of Shikata et

al. [16], whose model is also designed as an extension of traditional public-key signature security notions. We compare our model with [16] in Section 6.1.

The Hara et al. [9] model for unconditionally secure blind signatures is essentially the same as the Shikata et al. model with an added blindness condition. Hara et al. separate the unforgeability definition of [16] into a weaker notion of unforgeability and an additional non-repudiation requirement. The non-repudiation requirement actually treats more cases than a simple non-repudiation attack (as the success of the attack is not dependent on dispute resolution), so the reason for this separation is unclear. The authors of [9] also allow the signer to be the target verifier, which is not explicitly allowed in the Shikata et al. model, and so they add a separate unforgeability definition for this case.

The models of Hanaoka et al. [7, 8] and Safavi-Naini et al. [12] are based on security notions from message authentication codes (MACs). Hanaoka et al. treat only a limited attack scenario (which is covered by our model), including *impersonation*, *substitution*, and *transfer with a trap*, and they do not include a verification oracle. Safavi-Naini et al. treat a similar range of attacks as our model, specified through *denial*, *spoofing*, and *framing* attacks, and allow both signature and verification oracles. It is unclear whether Safavi-Naini et al. meant to ensure strong unforgeability, as the relationship between successful forgeries and oracle queries is unspecified. Furthermore, our model is more concise, as the denial attack covers a signer trying to repudiate a signature, whereas we show that it is unnecessary to treat non-repudiation as a separate part of an unforgeability definition. In addition, not all attack scenarios included in our definition are covered by the Safavi-Naini et al. model. For instance, the attack consisting of signer $U_\zeta \in C$ with target verifier U_v, where C creates a (ζ, v)-fraudulent signature, is not considered. The Safavi-Naini et al. model considers this scenario only in the case where an arbiter is involved and rejects the signature (i.e. a denial attack). In certain applications (e.g., e-cash) we do not want the signer to be able to create a (ζ, v)-fraudulent signature, regardless of whether a dispute resolution mechanism is invoked.

6.1 Comparison with the model of Shikata et al.

In this section, we discuss several aspects of the model of Shikata et al. [16] and how our approach differs from theirs.

1. Shikata et al.'s model [16] is limited to a single-signer scenario. We consider a more general model in which any participant can be a signer.
2. In [16, Definition 2], a signed message (x, σ) is defined to be *valid* if it was created using the signer's signing algorithm. Then, in their Requirement 1, which includes notions for verifiability, dispute resolution, and unforgeability, it is stated that (x, σ) is valid if and only if U_v's verification algorithm outputs *True* when given (x, σ) as input. This requirement is problematic, since U_v can use knowledge of his verification algorithm to find a pair (x, σ) that has output *True*; such a pair is then "valid." However, this means that a receiver can create valid signatures, and consequently the signature scheme does not provide unforgeability. Shikata et al. relax this condition in their Requirement 2 by allowing a small error probability that an "invalid" signature is accepted by a given verifier. However, this does not rectify the aforementioned problem, as the probability space in this definition is unspecified.
3. Shikata et al.'s definitions of *existential forgery* and *existential acceptance forgery* [16, Definitions 3 and 4] are rather complicated. It seems that the notion of "existential forgery" corresponds to our definition of an *ζ-authentic signature*. The coalition that creates this signature should not include U_ζ. The notion of "existential acceptance forgery" apparently is dependent upon the coalition that creates it. If U_ζ is in the coalition, then an existential acceptance forgery would most naturally coincide with our definition of an *(ζ, v)-fraudulent signature*. If U_ζ is not in the coalition, then it would more likely mean an *(ζ, v)-acceptable signature*. In each case, the coalition creating the signature should not include U_v. These definitions are a bit confusing, and we believe that the concepts of authentic, acceptable, and fraudulent signatures are helpful in phrasing clear and concise definitions.
4. In [16, Theorem 2], it is stated without proof that a signature scheme that is "existentially acceptance unforgeable" is necessarily "existentially unforgeable." Roughly speaking, this is logically equivalent

to the statement that an adversary that can create an existential forgery can also create an existential acceptance forgery. This statement seems rather obvious, but we need to also consider the coalitions that are creating these signatures. The adversary creating the existential forgery (i.e., a ζ-authentic signature) could be any coalition C that does not include U_ζ. A ζ-authentic signature is an existential acceptance forgery for any user $U_\nu \notin C \cup \{U_\zeta\}$. However, a problem arises if C consists of all users except for U_ζ. In this situation, a ζ-authentic signature created by C is not an existential acceptance forgery for any user. This situation is not accounted for in [16, Theorem 2], and therefore it does not suffice to consider only existential acceptance forgeries. We remark that our approach is consistent with that used to define A^2-codes [18], in which neither the sender nor the receiver is trusted, and so attacks solely against a target signer are considered. To be specific, Simmons [18] treats R_0 attacks, impersonation by the receiver, and R_1 attacks, substitution by the receiver. Allowing attacks in which all verifiers collude against a target signer is a generalization of this approach.

5. Notwithstanding the previous points, Shikata et al.'s definition of "strong security" [16, Definition 9] is very similar to our properties 4a and 4b of Definition 3.1, except that their Definition 9 only covers existential acceptance forgeries. In order to compare our model with that of Shikata et al. [16], we consider the following three attack scenarios, where U_ζ denotes the signer and U_ν denotes a verifier:
   Case A Neither U_ζ nor U_ν is in the coalition C, and C creates a (ζ, ν)-fraudulent signature.
   Case B U_ζ is not in the coalition C, and C creates a ζ-authentic signature.
   Case C $U_\zeta \in C$, $U_\nu \notin C$, and C creates a (ζ, ν)-fraudulent signature.
   In our security definition (Definition 3.1), property 4a is equivalent to the union of Case A and Case C, and property 4b is equivalent to Case B. Now, [16, Definition 9] considers two attacks: property 1) is the union of Cases A and B, but does not include the case where there is no target verifier, as discussed in the previous point; and property 2) is Case C.
6. Finally, we give a more complete treatment of dispute resolution than is presented by Shikata et al. [16].

7 Basic USS scheme construction and analysis

Current literature favors constructions using multivariate polynomials. We consider the security of the construction from Hanaoka et al. [7] in our security model. We reiterate that Hanaoka et al. [7] do not provide a proof of security for this construction in their model.

7.1 Key pair generation

Let $\mathbb{F}_q$ be a finite field with q elements such that $q > n$. (In practice, we pick q to be much larger than n.) The TI picks n *verification vectors* $\vec{v_1}, \ldots, \vec{v_n} \in (\mathbb{F}_q)^\omega$ uniformly at random for users $U_1, \ldots, U_n$, respectively, subject to one additional constraint. For technical reasons, we assume the verification vectors $\vec{v_1}, \ldots, \vec{v_n} \in (\mathbb{F}_q)^\omega$ satisfy the additional property that for any subset of size $\omega + 1$, the corresponding subset of size $\omega + 1$ formed from the new vectors $[1, \vec{v_1}], \ldots, [1, \vec{v_n}] \in (\mathbb{F}_q)^{\omega+1}$ is a linearly independent set. (This linear independence assumption is used in the security proof in Section 7.3.) We assume user identities $U_1, \ldots, U_n$ have a representation as elements in $\mathbb{F}_q$ in some suitable (and public) way.

The TI constructs the polynomial $F(x, y_1, \ldots, y_\omega, z)$ as

$$F(x, y_1, \ldots, y_\omega, z) = \sum_{i=0}^{n-1} \sum_{k=0}^{\psi} a_{i0k} x^i z^k + \sum_{i=0}^{n-1} \sum_{j=1}^{\omega} \sum_{k=0}^{\psi} a_{ijk} x^i y_j z^k,$$

where the coefficients $a_{ijk} \in \mathbb{F}_q$ are chosen uniformly at random.

For each U_ζ for $1 \le \zeta \le n$, the TI computes the *signing key*

$$s_\zeta(y_1, \ldots, y_\omega, z) = F(x, y_1, \ldots, y_\omega, z)|_{x=U_\zeta}$$

and the *verification key*

$$\tilde{v}_\zeta(x, z) = F(x, y_1, \ldots, y_\omega, z)|_{(y_1,\ldots,y_\omega)=\vec{v}_\zeta}.$$

For each $U_\zeta \in \mathcal{U}$, the TI distributes the corresponding verification vector $\vec{v}_\zeta$, signing key $s_\zeta(y_1, \ldots, y_\omega, z)$, and verification key $\tilde{v}_\zeta(x, z)$. It is assumed the TI can communicate with the users via secure channels and deletes the information afterwards.

7.2 Signature generation and verification

For a message $m \in \mathbb{F}_q$, a user U_ζ generates a signature σ by

$$\sigma(y_1, \ldots, y_\omega) = s_\zeta(y_1, \ldots, y_\omega, z)|_{z=m}.$$

To verify a signature pair (m, σ) from U_ζ, a user U_v checks that

$$\sigma(y_1, \ldots, y_\omega)|_{(y_1,\ldots,y_\omega)=\vec{v}_v} = \tilde{v}_v(x, z)|_{x=U_\zeta, z=m}.$$

Remark 7.1. The parameter ω in the construction determines the maximum number of colluders the scheme protects against and the parameter ψ determines the maximum number of signatures each user can produce without revealing their signing information. This is discussed in detail in the security analysis, but for clarity we briefly sketch how the construction relates to these bounds. In particular, each signing key s_ζ is a polynomial of degree ψ in z, so users cannot produce more than ψ signatures without revealing s_ζ. In addition, if a coalition C consists of $\omega + 1$ users or more, then the verification keys $\{\tilde{v}_h\}_{U_h \in C}$ suffice to reconstruct F. This follows because F is a linear polynomial in $\mathbb{F}_q[x, z][y_1, \ldots, y_\omega]$, and each verification key is a point on $F(y_1, \ldots, y_\omega)$. In this case the coalition has $\omega + 1$ linearly independent linear equations in $\omega + 1$ unknowns, and so C can solve for F.

7.3 Security analysis

Given q, we define the security parameter to be k, where $k = \log_2 q$. We consider the game Sig-forge$_{C,\Pi}(k)$ and calculate the probability that the output is 1. In particular, we consider the probability that the coalition C produces a signature pair (m, σ) satisfying Conditions 4a and 4b of Definition 3.1 separately. Here we prove the scheme is unforgeable with respect to coalitions C of size at most ω, where C is allowed $\psi_S = (n - \omega)\psi$ oracle queries to $\mathsf{Sign}^{\mathcal{O}}$ (where ψ is the total number of $\mathsf{Sign}_h^{\mathcal{O}}$ oracle queries allowed for each user $U_h \notin C$), and where the number of $\mathsf{Vrfy}^{\mathcal{O}}$ queries, say ψ_V, is arbitrary. (As shown in the following theorem, the probability that C creates a successful forgery depends on this value ψ_V.) That is, we allow C to have at most ω members and to have access to ψ sample signatures from each user $U_h \notin C$. (This is consistent with the fact that in this USS scheme, each user is allowed to produce at most ψ signatures, so the bound on oracle access to $\mathsf{Sign}_h^{\mathcal{O}}$ for each user $U_h \notin C$ must be ψ.)

Theorem 7.2. *Under the above assumptions, C outputs a signature pair (m, σ) in the game* Sig-forge$_{C,\Pi}(k)$ *of Definition 3.1 satisfying Condition 4a or 4b with probability at most $\frac{r}{1-r}$, where $r = \frac{\psi_V}{q-1}$.*

Proof. Recall the assumption that the verification vectors $\vec{v}_1, \ldots, \vec{v}_n \in (\mathbb{F}_q)^\omega$ satisfy the additional property that for any subset of size $\omega + 1$, the corresponding subset of size $\omega + 1$ formed from the new vectors $[1, \vec{v}_1], \ldots, [1, \vec{v}_n] \in (\mathbb{F}_q)^{\omega+1}$ is a linearly independent set. We use this fact throughout the proof.

We wish to consider the strongest possible coalition C. To this end, we consider a coalition of size ω whose verification vectors form a linearly independent set. Lemma A.1 implies that this is always possible. Without loss of generality, assume our adversaries are $C = \{U_1, \ldots, U_\omega\}$, with target signer U_ζ and target verifier U_v. The coalition C outputs a signature pair (m, σ) with claimed signer U_ζ.

For ease of notation, define $y_0 = 1$ and let $\vec{y}$ denote the vector $(y_0, y_1, \ldots, y_\omega)$. Then we have

$$F(x, \vec{y}, z) = \sum_{i=0}^{n-1} \sum_{j=0}^{\omega} \sum_{k=0}^{\psi} a_{ijk} x^i y_j z^k.$$

We sometimes refer to the user U_h's augmented verification vector, namely $[1, \vec{v_h}] = (1, v_{h,1}, \ldots, v_{h,\omega})$, as $(v_{h,0}, \ldots, v_{h,\omega})$.

The polynomial F is determined by the $n(\omega+1)(\psi+1)$ unknown coefficients a_{ijk}. The coalition C has access to the following information:

1. The verification keys $\tilde{v}_1, \ldots, \tilde{v}_\omega$. We have, for $U_h \in C$,

$$\tilde{v}_h(x,z) = F(x,\vec{y},z)|_{\vec{y}=\vec{v_h}} = \sum_{i=0}^{n-1}\sum_{j=0}^{\omega}\sum_{k=0}^{\psi} a_{ijk}x^i v_{h,j} z^k.$$

   Noting that $\tilde{v}_h$ is a polynomial with terms of the form $(c_{ik})_h x^i z^k$ for $0 \le i \le n-1$ and $0 \le k \le \psi$, we see that the coalition C has access to $n(\psi+1)(\omega)$ equations C_{ikh} in the unknowns a_{ijk}, where

$$C_{ikh}\colon\ a_{i0k} + \sum_{j=1}^{\omega} a_{ijk} v_{h,j} = (c_{ik})_h$$

   for some (known) element $(c_{ik})_h \in \mathbb{F}_q$.
   We note that these equations

$$\{C_{ikh} : 0 \le i \le n-1,\ 0 \le k \le \psi,\ 1 \le h \le \omega\} \tag{7.1}$$

   form a linearly independent set, since the rank of $\{(1, \vec{v_1}), \ldots, (1, \vec{v_\omega})\} \subseteq (\mathbb{F}_q)^{\omega+1}$ is ω. More details are provided in Appendix A.1.

2. The signing keys $s_1, \ldots, s_\omega$. We have, for $U_h \in C$,

$$s_h(\vec{y},z) = F(x,\vec{y},z)|_{x=U_h} = \sum_{i=0}^{n-1}\sum_{j=0}^{\omega}\sum_{k=0}^{\psi} a_{ijk}U_h^i y_j z^k.$$

   Noting that s_h is a polynomial with terms of the form $(d_{jk})_h y_j z^k$, for $0 \le j \le \omega$ and $0 \le k \le \psi$, we have that C has access to $(\omega+1)(\psi+1)(\omega)$ equations D_{jkh} in the unknowns a_{ijk}, where

$$D_{jkh}\colon\ \sum_{i=0}^{n-1} a_{ijk}U_h^i = (d_{jk})_h$$

   for some (known) element $(d_{jk})_h \in \mathbb{F}_q$.
   Now, these equations, together with the equations from (7.1), are not a linearly independent set, due to the relationships between users' signing and verification keys. More specifically, for any users U_h and $U_{h'}$, we have

$$s_h(\vec{y},z)|_{\vec{y}=\vec{v_{h'}}} = \tilde{v}_{h'}(x,z)|_{x=U_h}. \tag{7.2}$$

   Equation (7.2) implies that for each $U_{h'} \in C$ and each choice of $0 \le k \le \psi$, we have a set of ω relations among the $\omega+1$ equations $\{D_{jkh} : 0 \le j \le \omega\}$.
   Thus, the information gleaned from the coalition's signing information is contained in the set

$$\{D_{jkh} : 0 \le k \le \psi,\ 1 \le h \le \omega\}. \tag{7.3}$$

3. Up to ψ signatures $\sigma_{h,k'}$ from each user $U_h \notin C$, on messages $m_{h,k'}$ of C's choice, where $1 \le k' \le \psi$, with the exception that C can only access a signature $\sigma_{\zeta,k'}$ on a message $m_{\zeta,k'} \ne m$ with target signer U_ζ. Thus C has access to $n-\omega$ signatures of the form

$$\sigma_{h,k'}(\vec{y}) = s_h(\vec{y},z)|_{z=m_{h,k'}} = \sum_{i=0}^{n-1}\sum_{j=0}^{\omega}\sum_{k=0}^{\psi} a_{ijk}U_h^i y_j (m_{h,k'})^k.$$

   Note that $\sigma_{h,k'}$ is a polynomial with terms of the form $(b_j)_{h,k'} y_j$. Then C has access to $(\omega+1)(\psi)(n-\omega)$ equations $B_{jhk'}$ in the unknowns a_{ijk}, where

$$B_{jhk'}\colon\ \sum_{i=0}^{n-1}\sum_{k=0}^{\psi} a_{ijk}U_h^i (m_{h,k'})^k = (b_j)_{h,k'}$$

   for some (known) element $(b_j)_{h,k'} \in \mathbb{F}_q$.

In a manner similar to the above analysis, we observe that

$$\sigma_{h,k'}(\vec{y})|_{\vec{y}=\vec{v_{h'}}} = \tilde{v}_{h'}(x,z)|_{x=U_h,z=m_{h,k'}}$$

for each $U_{h'} \in C$. Thus it suffices to consider the set

$$\{B_{0hk'} : 1 \le k' \le \psi\} \tag{7.4}$$

for each $U_h \notin C$.

4. Up to ψ_V query results from the oracle $\mathsf{Vrfy}_h^{\mathcal{O}}$ for $U_h \notin C$. In the following, we first consider the attack scenario without $\mathsf{Vrfy}^{\mathcal{O}}$ queries and then move to incorporate these queries into the analysis.

To summarize, the information obtained by C is contained in equation sets (7.1) and (7.3), together with, for each $U_h \notin C$, equation set (7.4). These equations form a linearly independent set; we provide the proof in Appendix A.1. We have a total of $n\omega\psi + n\omega + \omega + \psi n$ equations, which implies we have $n - \omega$ free variables in the given linear system.

With the given information, C can consider the polynomials $F'(x, \vec{y}, z)$ consistent with the known information about $F(x, \vec{y}, z)$. If a given polynomial F' is consistent with the known information about F, we say F' satisfies property $(*)$. We let

$$\mathcal{F} = \{F'(x, \vec{y}, z) : F' \text{ satisfies } (*)\}.$$

From above, we have $|\mathcal{F}| = q^{n-\omega}$.

Case 1: $U_\zeta \notin C$, $U_v \in C$. In this case, the goal of C is to produce a ζ-authentic signature; we wish to give an upper bound on C's probability of success, so we consider the most advantageous method by which C can create such a signature. If C creates a ζ-authentic signature (m, σ) consistent with C's known information, then this is equivalent to C finding U_ζ's signing key $s_\zeta(\vec{y}, z)$. This follows because C would then have access to $\psi + 1$ points $\sigma(\vec{y}), \sigma_{\zeta,1}(\vec{y}), \ldots, \sigma_{\zeta,\psi}(\vec{y})$ on $s_\zeta(\vec{y}, z)$, which is a polynomial of degree ψ in z.

The above observation implies we can calculate the probability of success as

$$\frac{\left|\{F'(x, \vec{y}, z) \in \mathcal{F} : F'(U_\zeta, \vec{y}, z) = F(U_\zeta, \vec{y}, z)\}\right|}{\left|\{F'(x, \vec{y}, z) \in \mathcal{F}\}\right|}.$$

Using the same notation as before, if $F'(U_\zeta, \vec{y}, z) = F(U_\zeta, \vec{y}, z)$, we have the $\psi + 1$ additional equations $\{D_{0k\zeta} : 0 \le k \le \psi\}$, rendering the equations $\{B_{0k'\zeta} : 1 \le k' \le \psi\}$ redundant. We can show the resulting set is linearly independent, so we have one additional restriction on F'. Recalling that we chose F' from a space of size $q^{n-\omega}$ initially, the coalition C's probability of success is

$$\frac{q^{n-\omega-1}}{q^{n-\omega}} = \frac{1}{q}.$$

Now, suppose C also has access to the $\mathsf{Vrfy}^{\mathcal{O}}$ oracle. We observe that if the query (m, σ) to $\mathsf{Vrfy}_{h'}^{\mathcal{O}}$ results in *True* (for some $U_{h'} \notin C$), and (m, σ) is consistent with C's information about F, then C has successfully determined U_ζ's signing key $s_\zeta(\vec{y}, z)$. To see this, first note that if (m, σ) is consistent with C's information about F, then $\sigma(\vec{y}) = F'(x, \vec{y}, z)|_{x=U_\zeta,z=m}$ for some $F' \in \mathcal{F}$. This implies $F'(x, \vec{y}, z)|_{x=U_\zeta,z=m}$ agrees with $F(x, \vec{y}, z)|_{x=U_\zeta,z=m}$ on the $\omega + 1$ points $\vec{v_1}, \ldots, \vec{v_\omega}, \vec{v_{h'}}$. By assumption, the augmented verification vectors $[1, \vec{v_1}], \ldots, [1, \vec{v_\omega}]$, $[1, \vec{v_{h'}}] \in (\mathbb{F}_q)^{\omega+1}$ are linearly independent, so we have

$$F'(x, \vec{y}, z)|_{x=U_\zeta,z=m} = F(x, \vec{y}, z)|_{x=U_\zeta,z=m}.$$

In other words, (m, σ) is a ζ-authentic signature. (This result is a consequence of basic linear algebra; we provide the relevant theory in Lemma A.2 of the Appendix.)

Now, any $F' \in \mathcal{F}$ also satisfies

$$F'(x, \vec{y}, z)|_{x=U_\zeta,z=m_{\zeta,k'}} = F(x, \vec{y}, z)|_{x=U_\zeta,z=m_{\zeta,k'}}$$

for $1 \le k' \le \psi$ and distinct messages $m_{\zeta,k'} \ne m$. That is, we have a total of $\psi + 1$ points at which $F'(x, \vec{y}, z)|_{x=U_\zeta}$ and $F(x, \vec{y}, z)|_{x=U_\zeta}$ agree as polynomials in z. Since F' and F are polynomials of degree ψ in z, this is sufficient

to conclude

$$F'(x, \vec{y}, z)|_{x=U_\zeta} = F(x, \vec{y}, z)|_{x=U_\zeta} = s_\zeta(\vec{y}, z),$$

as desired. The probability of C finding s_ζ, however, is the probability of C choosing the correct F', which, as we show below, is $\frac{1}{q-\psi_V}$, where ψ_V is the number of queries to $\mathsf{Vrfy}^{(\cdot)}$ with result *False*.

We now consider ψ_V queries to $\mathsf{Vrfy}^{(\cdot)}$ with result *False*, supposing each query is consistent with C's view of the function F. We observe that each negative query eliminates (at most) one potential signing key for U_ζ. Given that the condition for success does not depend on the particular target verifier's verification key $\vec{v}_v$, we can calculate the probability of success as before, this time allowing for information gleaned from the ψ_V negative queries. We write $\bar{s}_\zeta^1, \ldots, \bar{s}_\zeta^{\psi_V}$ for these eliminated signing keys, and for readability, we write $F'_\zeta(\vec{y}, z)$ for $F'(x, \vec{y}, z)|_{x=U_\zeta}$.

We first need to calculate the number of possible functions F' consistent with C's view of F, that is, $|\{F'(x, \vec{y}, z) \in \mathcal{F} : F'_\zeta \notin \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\}\}|$. We have

$$\left|\{F' \in \mathcal{F} : F'_\zeta \notin \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\}\}\right| = \left|\{F' \in \mathcal{F}\}\right| - \left|\{F' \in \mathcal{F} : F'_\zeta \in \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\}\}\right|.$$

We assume the events $F'_\zeta = \bar{s}_\zeta^1, \ldots, F'_\zeta = \bar{s}_\zeta^{\psi_V}$ are disjoint, since if $\bar{s}_\zeta^i = \bar{s}_\zeta^j$ for some $1 \le i, j \le \psi_V$, this is equivalent to fewer verification oracle queries. Following the same reasoning as before, we then have

$$\begin{aligned}\left|\{F' \in \mathcal{F} : F'_\zeta \notin \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\}\}\right| &= \left|\{F' \in \mathcal{F}\}\right| - \sum_{i=1}^{\psi_V} \left|\{F' \in \mathcal{F} : F'_\zeta = \bar{s}_\zeta^i\}\right| \\ &= q^{n-\omega} - \psi_V q^{n-\omega-1} \\ &= q^{n-\omega-1}(q - \psi_V).\end{aligned}$$

We can calculate C's probability of success on the ψ_Vth oracle query as

$$\begin{aligned}\frac{\left|\{F'(x, \vec{y}, z) \in \mathcal{F} : F'_\zeta \notin \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\},\ F'_\zeta = s_\zeta\}\right|}{\left|\{F'(x, \vec{y}, z) \in \mathcal{F} : F'_\zeta \notin \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\}\}\right|} &= \frac{\left|\{F'(x, \vec{y}, z) \in \mathcal{F} : F'_\zeta = s_\zeta\}\right|}{\left|\{F'(x, \vec{y}, z) \in \mathcal{F} : F'_\zeta \notin \{\bar{s}_\zeta^1, \bar{s}_\zeta^2, \ldots, \bar{s}_\zeta^{\psi_V}\}\}\right|} \\ &= \frac{q^{n-\omega-1}}{q^{n-\omega-1}(q - \psi_V)} \\ &= \frac{1}{q - \psi_V}.\end{aligned}$$

C's overall probability of success in this case is then $\sum_{i=0}^{\psi_V} \frac{1}{q-i}$.

Case 2: $U_\zeta \notin C$, $U_v \notin C$. Now suppose $U_v \notin C$. In this case, the goal of C is to produce a (ζ, v)-acceptable signature. Note that in order for a signature pair (m, σ) with claimed signer U_ζ to pass U_v's verification check, (m, σ) must satisfy

$$\sigma(\vec{y})|_{\vec{y}=\vec{v}_v} = \tilde{v}_V(x, z)|_{x=U_\zeta, z=m}.$$

In particular, if (m, σ) is consistent with both U_v's verification key and with F, then the same analysis as in the previous case implies that (m, σ) is a ζ-authentic signature, and indeed that C has determined s_ζ. Thus, the set of known information $(*)$ does not help create a (ζ, v)-fraudulent signature. For the case of creating a (ζ, v)-fraudulent signature, the most powerful collusion C includes the signer U_ζ, which we consider next.

Case 3: $U_\zeta \in C$, $U_v \notin C$. Here C's goal is to produce a (ζ, v)-fraudulent signature. Since the polynomial F and the set of verification vectors $\vec{v}_1, \ldots, \vec{v}_n \in (\mathbb{F}_q)^\omega$ are chosen independently, we see that the signing keys of C and sample signatures from $U_h \notin C$ have no bearing on the probability distribution for the key $\vec{v}_v$.

Recall that for any subset of the set $\{\vec{v}_1, \ldots, \vec{v}_n\}$ of size $\omega + 1$, the corresponding subset of size $\omega + 1$ formed from the new vectors $[1, \vec{v}_1], \ldots, [1, \vec{v}_n] \in (\mathbb{F}_q)^{\omega+1}$ is a linearly independent set. Therefore, knowledge of the keys $\vec{v}_h$ for $U_h \in C$ does affect the probability distribution for the key $\vec{v}_v$. In particular, C is aware that $[1, \vec{v}_v] \neq \sum_{j=1}^{\omega} k_j [1, \vec{v}_j]$ for any choice of $\{k_1, \ldots, k_\omega \in \mathbb{F}_q : \sum_{j=1}^{\omega} k_j = 1\}$. That is, given $\vec{v}_1, \ldots, \vec{v}_\omega$, there are $q^\omega - q^{\omega-1}$ choices for $\vec{v}_v$, any of which are equally likely. We write V for the set of possible vectors $\vec{v}_v$.

Now suppose we want to create a (ζ, ν)-fraudulent signature $\sigma'(\vec{y})$ on a message m. Suppose $\sigma(\vec{y}) = b_0 + \sum_{j=1}^{\omega} b_j y_j$ is the ζ-authentic signature on m. Then writing

$$\sigma'(\vec{y}) = b'_0 + \sum_{j=1}^{\omega} b'_j y_j,$$

we need $\sigma(\vec{v}_\nu) = \sigma'(\vec{v}_\nu)$, but $(b_0, \ldots, b_\omega) \neq (b'_0, \ldots, b'_\omega)$.

In other words, C needs to find a nonzero vector $\vec{\beta} = (b_0 - b'_0, \ldots, b_\omega - b'_\omega)$ satisfying $\vec{\beta} \cdot [1, \vec{v}_\nu] = 0$. The probability of success is then calculated as

$$\max_{\vec{\beta}} \frac{\left|\{\vec{v}_\nu \in V : \vec{\beta} \cdot [1, \vec{v}_\nu] = 0\}\right|}{\left|\{\vec{v}_\nu \in V\}\right|} \leq \max_{\vec{\beta}} \frac{\left|\{\vec{v}_\nu \in (\mathbb{F}_q)^\omega : \vec{\beta} \cdot [1, \vec{v}_\nu] = 0\}\right|}{\left|\{\vec{v}_\nu \in V\}\right|} = \frac{q^{\omega-1}}{q^\omega - q^{\omega-1}} = \frac{1}{q-1}.$$

We now consider $\mathsf{Vrfy}^{\mathcal{O}}$ queries. We observe that a positive $\mathsf{Vrfy}_\nu^{\mathcal{O}}$ query (m, σ) allows the coalition C to win the game $\text{Sig-forge}_{C,\Pi}(k)$, so we consider the probability of success given ψ_V negative $\mathsf{Vrfy}_\nu^{\mathcal{O}}$ queries, since this gives the best chance of success. (Note that it is also possible, albeit extremely unlikely, that a positive $\mathsf{Vrfy}_\nu^{\mathcal{O}}$ query here results in a forgery that is ζ-authentic. The coalition C also wins in this instance, but we are not concerned with ζ-authentic forgeries here, as the best approach to producing these types of forgeries is analyzed in Case 1.)

We let V' be the set of possible vectors $\vec{v}_\nu$ given the new knowledge gleaned from the ψ_V negative query vectors $\vec{\beta}_1, \ldots, \vec{\beta}_{\psi_V}$. That is,

$$V' = \{\vec{v}_\nu \in V : \vec{\beta}_1 \cdot [1, \vec{v}_\nu] \neq 0, \ldots, \vec{\beta}_{\psi_V} \cdot [1, \vec{v}_\nu] \neq 0\}.$$

Now,

$$\begin{aligned}
\left|\{\vec{v}_\nu \in V'\}\right| &= \left|\{\vec{v}_\nu \in V\}\right| - \left|\{\vec{v}_\nu \in V : \vec{\beta}_1 \cdot [1, \vec{v}_\nu] = 0 \text{ or } \cdots \text{ or } \vec{\beta}_{\psi_V} \cdot [1, \vec{v}_\nu] = 0\}\right| \\
&\geq \left|\{\vec{v}_\nu \in V\}\right| - \left|\{\vec{v}_\nu \in (\mathbb{F}_q)^\omega : \vec{\beta}_1 \cdot [1, \vec{v}_\nu] = 0 \text{ or } \cdots \text{ or } \vec{\beta}_{\psi_V} \cdot [1, \vec{v}_\nu] = 0\}\right| \\
&\geq \left|\{\vec{v}_\nu \in V\}\right| - \sum_{i=1}^{\psi_V} \left|\{\vec{v}_\nu \in (\mathbb{F}_q)^\omega : \vec{\beta}_i \cdot [1, \vec{v}_\nu] = 0\}\right| \\
&= (q^\omega - q^{\omega-1}) - \psi_V q^{\omega-1} \\
&= q^{\omega-1}(q - \psi_V - 1).
\end{aligned}$$

C's probability of success on the ψ_Vth oracle query is then

$$\begin{aligned}
\max_{\vec{\beta}} \frac{\left|\{\vec{v}_\nu \in V' : \vec{\beta} \cdot [1, \vec{v}_\nu] = 0\}\right|}{\left|\{\vec{v}_\nu \in V'\}\right|} &\leq \max_{\vec{\beta}} \frac{\left|\{\vec{v}_\nu \in (\mathbb{F}_q)^\omega : \vec{\beta} \cdot [1, \vec{v}_\nu] = 0\}\right|}{\left|\{\vec{v}_\nu \in V'\}\right|} \\
&\leq \frac{q^{\omega-1}}{q^{\omega-1}(q - \psi_V - 1)} = \frac{1}{q - \psi_V - 1}.
\end{aligned}$$

This implies the probability of success (considering all verification oracle queries) in this case is $\sum_{i=0}^{\psi_V} \frac{1}{q-i-1}$.

As Case 3 treats the most powerful coalition, we use these results to establish simple bounds for the overall probability of success. Setting $r = \frac{\psi_V}{q-1}$, we have

$$r \leq \text{Sig-forge}_{C,\Pi}(k) \leq \frac{r}{1-r}.$$

This completes the proof of Theorem 7.2. □

Remark 7.3. The linear independence assumption in the above construction is not necessary, as observed by Hanaoka et al. [7], but it does simplify the security analysis. If the linear independence assumption is not satisfied, we must take into account the rank of $\{\vec{v}_h : U_h \in C\}$, which may be strictly less than ω. In this case, the coalition C has less information, but the proof is similar. We can also increase the robustness of the construction against verification oracle queries by using a polynomial $F(x, y_1, \ldots, y_{\omega+\tau}, z)$ of the same form as above, where $\tau > 0$. This achieves security as outlined in Theorem 7.2, where the coalition has, in addition, achieved up to τ successful verification oracle queries. This technique is used by Shikata et al. [16] in their construction, although it is not explained.

8 USS schemes with key insulation

Key exposure is a major concern in any cryptosystem. In traditional public-key cryptography, Dodis et al. [5] introduced the notion of *key insulation*, in which a user's secret information is split between a physically secure (and perhaps computationally limited) device H, and an insecure device with temporary secret keys that are refreshed at intervals with information sent by H. These notions have been applied to signatures in the traditional setting by Dodis et al. [6] and to unconditionally secure multi-receiver authentication codes and key agreement by Seito et al. [14] and Seito and Shikata [15]. In this section, we concern ourselves with *key exposure of a user's signing information*. Our main goal is to provide an example of how our basic USS security model might be extended to incorporate more complicated security notions, such as key insulation. Our definitions are extensions of those provided by Seito et al. [14] and Seito and Shikata [15] to the signature setting, keeping in mind the original goals of Dodis et al. [6].

The basic idea is as follows. A user's signing information is split into a "master" signing key stored on a secure device, temporary secret-signing keys, which are derived from an initial secret (stored on an insecure device), and key-updating information (which is sent at intervals from the secure device). For each signer, we want the scheme to be robust against exposure of *either* that user's master signing key *or* some (strict) subset of the user's temporary signing keys, *but not both*. The overall scheme should be secure provided these exposure criteria hold for all honest users; this property is called *strong key insulation*.

We begin by providing a formal definition of an unconditionally secure signature scheme with key insulation (KI-USS) in Section 8.1. In Section 8.2, we give an extension of our basic security model from Section 3 to the key-insulation setting. Once we have established our formal security notions and model, we give an extension to the USS construction from Hanaoka et al. [7] (which we analyzed in Section 7) in Section 9. The extension is inspired by a multi-receiver authentication code construction presented by Seito et al. [14].

8.1 Preliminary definitions

We require the following definitions.

Definition 8.1. An *unconditionally secure signature scheme with key insulation* (or *KI-USS scheme*) Π consists of a trusted initializer TI, a set $\mathcal{U}$ of n users, a set $\mathcal{H}$ of n secure devices, a tuple of seven spaces $(\mathcal{T}, \mathcal{X}, \Sigma, \mathcal{V}, \mathcal{I}, \mathcal{MK}, \mathcal{SK})$, and algorithms Gen and $\{\mathsf{Sign}_\zeta, \mathsf{Vrfy}_\zeta, \mathsf{MKUpd}_\zeta, \mathsf{SKUpd}_\zeta\}_{1\le\zeta\le n}$, satisfying the following:

- The set $\mathcal{U} = \{U_1, \ldots, U_n\}$ consists of n possible users.
- $\mathcal{H} = \{H_1, \ldots, H_n\}$ is a set of n secure devices, where each $H_i \in \mathcal{H}$ is the secure device for user $U_i \in \mathcal{U}$.
- $\mathcal{T} = \{0, 1, 2, \ldots, N\}$ is a set of time periods.
- $\mathcal{X}$ is a finite set of possible messages.
- Σ is a finite set of possible signatures.
- $\mathcal{V}$ is a finite set of possible (secret) verification information.
- $\mathcal{I}$ is a finite set of possible secret-key-updating information (to keep track of time periods).
- $\mathcal{MK}$ is a finite set of possible (secret) master keys.
- $\mathcal{SK}$ is a finite set of possible secret signing keys. The set $\mathcal{SK}^t$ is the set of possible signing keys at time period t.
- The *key-generation algorithm* Gen takes as input 1^k, where k is a security parameter, and the total number of time periods N and outputs a master secret key $mk^* := (mk_1, \ldots, mk_n) \in \mathcal{MK}^n$ and initial signing key information $sk^* := (sk_1^0, \ldots, sk_n^0) \in \mathcal{SK}^n$, together with verification keys $\{v_\zeta \in \mathcal{V} : 1 \le \zeta \le n\}$.
- For each $U_\zeta \in \mathcal{U}$, the *master-key-updating algorithm* $\mathsf{MKUpd}_\zeta : \mathcal{MK} \times \mathcal{T} \to \mathcal{I}$ for user U_ζ takes as input U_ζ's master key mk_ζ, and a time period $t \in T$, and returns secret key-updating information $mk_\zeta^{(t-1,t)} \in \mathcal{I}$. The key-updating information $mk_\zeta^{(t-1,t)} \in \mathcal{I}$ is used by U_ζ in order to update his signing key from time period $t-1$ to time period t, as described by the next algorithm SKUpd_ζ.

- For each $U_\zeta \in \mathcal{U}$, the *signing-key-updating algorithm* SKUpd_ζ: $\mathcal{T} \times \mathcal{SK} \times \mathcal{I} \to \mathcal{SK}$ takes as input a time period $t \in \mathcal{T}$, a secret signing key $sk_\zeta^{(t-1)}$ for time period $t-1$, and secret key-updating information $mk_\zeta^{(t-1,t)}$, and returns a signing key $sk_\zeta^t \in \mathcal{SK}^t$ for time period t.
- For each $U_\zeta \in \mathcal{U}$, the *signing algorithm* Sign_ζ: $\mathcal{T} \times \mathcal{X} \times \mathcal{SK} \to \Sigma$ takes as input a time period $t \in \mathcal{T}$ satisfying $t > 0$, a message $x \in \mathcal{X}$, and a signing key $sk_\zeta^t \in \mathcal{SK}$, and returns a signature $\sigma \in \Sigma$. We let Sign_ζ^t denote the algorithm $\mathsf{Sign}_\zeta(t, \cdot, sk_\zeta^t)$.
- For each $U_\zeta \in \mathcal{U}$, the *verification algorithm* Vrfy_ζ: $\mathcal{X} \times \mathcal{T} \times \Sigma \times \mathcal{U} \times \mathcal{V} \to \{True, False\}$ takes as input a message $x \in \mathcal{X}$, a time period $t \in \mathcal{T}$, a signature $\sigma \in \Sigma$, a signer $U_\nu \in \mathcal{U}$, and verification key $v_\zeta \in \mathcal{V}$, and outputs either *True* or *False*. For each user U_ζ, we let Vrfy_ζ^t denote the algorithm $\mathsf{Vrfy}_\zeta(\cdot, t, \cdot, \cdot, v_\zeta)$.

Scheme Phases:

1. *Key generation phase*. For each $U_\zeta \in \mathcal{U}$, the TI runs Gen and securely distributes $(mk_\zeta, sk_\zeta^0, v_\zeta)$ to U_ζ. The TI then deletes all keys from his memory. Each user U_ζ places his master key mk_ζ on his secure device H_ζ and then deletes mk_ζ from his memory.
2. *Update phase*. To update signing information for a user U_ζ from time period $t-1$ to period t, the secure device H_ζ runs $\mathsf{MKUpd}_\zeta(mk_\zeta, t)$ and sends the output $mk_\zeta^{(t-1,t)}$ to U_ζ via a secure channel. The user U_ζ then runs $\mathsf{SKUpd}(t, sk_\zeta^{(t-1)}, mk^{(t-1,t)})$, which outputs sk_ζ^t, the signing key for the new time period t. The user U_ζ then deletes $sk_\zeta^{(t-1)}$ and $mk^{(t-1,t)}$ from his memory.
3. *Signing phase*. To sign a message $x \in \mathcal{X}$ during a time period t, a user U_ζ runs his signing algorithm $\mathsf{Sign}_\zeta^t(x)$, which outputs a signature $\sigma \in \Sigma$. The user U_ζ then forms the signature triple (x, t, σ).
4. *Verification phase*. To verify a signature triple (x, t, σ) from a signer U_ν, a user U_ζ runs his verification algorithm $\mathsf{Vrfy}_\zeta^t(x, \sigma, U_\nu)$. If the output of $\mathsf{Vrfy}_\zeta^t(x, \sigma, U_\nu)$ is *True*, then U_ζ believes that the signature pair was actually produced by U_ν's signing algorithm during time period t as claimed.

It is required that, for every k, for every N, for every set $\{\mathsf{Sign}_\zeta, \mathsf{Vrfy}_\nu : 1 \le \zeta, \nu \le n\}$ of signing and verification algorithms output by $\mathsf{Gen}(1^k, N)$, for every pair $U_\zeta, U_\nu \in \mathcal{U}$, and for every $x \in \mathcal{X}$ and $t \in \mathcal{T}$ such that $t > 0$, it holds that

$$\mathsf{Vrfy}_\nu^t(x, \mathsf{Sign}_\zeta^t(x), U_\zeta) = True.$$

Remark 8.2. We are treating *deterministic* signature schemes only, in the sense that all algorithms except Gen are deterministic, although the above definition can easily be extended to the randomized setting.

8.2 Security model

The concepts of authentic, acceptable, and fraudulent signatures are defined as before.

Definition 8.3. A signature $\sigma \in \Sigma$ on a message $x \in \mathcal{X}$ during a time period $t \in \mathcal{T}$ is ζ-*authentic* if $\sigma = \mathsf{Sign}_\zeta^t(x)$.

Definition 8.4. A signature $\sigma \in \Sigma$ on a message $x \in \mathcal{X}$ during a time period $t \in \mathcal{T}$ is (ζ, ν)-*acceptable* if $\mathsf{Vrfy}_\nu^t(x, \sigma, U_\zeta) = True$.

Definition 8.5. A signature $\sigma \in \Sigma$ on a message $x \in \mathcal{X}$ during a time period $t \in \mathcal{T}$ is (ζ, ν)-*fraudulent* if σ is (ζ, ν)-acceptable but not ζ-authentic.

Informally, we wish to guard against two types of possible key exposure for each honest user U_ℓ. We want the scheme to be secure against *either* (but not both) of the following attacks on honest users $U_\ell \in \mathcal{U}$:

- *Signing key exposure*: Compromise of user U_ℓ's signing keys from the insecure device for up to γ time periods
- *Master key exposure*: Compromise of U_ℓ's secure device, where mk_ℓ is stored.

We need to define the following oracles. The first two of these oracles are used to model possible key exposure for honest users, and the latter two are direct generalizations of the signing and verification oracles used for regular USS schemes.

- The SigningExposure$^{\mathcal{O}}(\cdot, \cdot)$ *oracle*; this oracle takes as input a user $U_\ell \in \mathcal{U}$ and a time period $t \in \mathcal{T}$ (where $t > 0$) and outputs U_ℓ's signing information for period t, namely $\mathsf{Sign}_\ell^t(\cdot)$. This oracle is used to model

compromise of U_ℓ's insecure device for up to γ time periods, where U_ℓ's temporary signing keys are stored.

- The MasterExposure$^{\mathcal{O}}(\cdot)$ oracle; this oracle takes as input a user $U_\ell \in \mathcal{U}$ and outputs U_ℓ's master key mk_ℓ. This oracle is used to model compromise of U_ℓ's secure device, where the master key mk_ℓ is stored.
- The $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot,\cdot)$ *oracle*; this oracle takes as input a message $x \in \mathcal{X}$ and time period $t \in \mathcal{T}$ and outputs an ℓ-authentic signature on the message x for time t.
- The $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot,\cdot,\cdot,\cdot)$ *oracle*; this oracle takes as input a signature triple (x, t, σ) (i.e., a message $x \in \mathcal{X}$, a time period $t \in \mathcal{T}$, and a signature $\sigma \in \Sigma$) and a signer U_ζ, and runs user U_ℓ's verification algorithm on input (x, t, σ, U_ζ), outputting *True* or *False*.

We now define the formal model as follows:

Definition 8.6. Let Π be a KI-USS scheme (with notation as defined in Definition 8.1), with security parameter k. Let $C \subseteq \mathcal{U}$ be a coalition of at most ω users and let ψ_S, ψ_V, and γ be positive integers. We define the following *signature game* KI-Sig-forge$_{C,\Pi}(k)$ with target signer U_ζ and verifier U_v:

1. $\mathsf{Gen}(1^k)$ is run to obtain the pair (mk^*, sk^*).
2. The coalition C is given bounded access to the MasterExposure $\mathcal{O}(\cdot)$ and SigningExposure $\mathcal{O}(\cdot,\cdot)$ oracles, as well as the $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot,\cdot)$ and $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot,\cdot,\cdot,\cdot)$ oracles. The rules for oracle access are as follows:
   a. For each $U_\ell \notin C$, the coalition C is permitted *only one* of the following:
      - SigningExposure$^{\mathcal{O}}(U_\ell, t)$ for up to γ time periods $t \in \mathcal{T}\backslash\{0\}$;
      - MasterExposure$^{\mathcal{O}}(U_\ell)$.

      We let $T' = \{t \in \mathcal{T}\colon C \text{ has accessed SigningExposure}^{\mathcal{O}}(U_\zeta, t)\}$.
   b. The coalition C is given bounded access to the oracles $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot,\cdot)$ and $\mathsf{Vrfy}_\ell^{\mathcal{O}}(\cdot,\cdot,\cdot,U_\zeta)$ for ℓ satisfying $U_\ell \notin C$. In particular, C is allowed a total of ψ_S and ψ_V queries to the $\mathsf{Sign}^{\mathcal{O}}$ and $\mathsf{Vrfy}^{\mathcal{O}}$ oracles, respectively, with at most $\psi_S/(n - |C|)$ queries to $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot)$ for each ℓ satisfying $U_\ell \notin C$. It should be noted that C has unlimited access to the signing and verification algorithms of any $U_\ell \in C$. For each time period $t \in \mathcal{T}$, we let $\mathcal{Q}_t$ denote the set of messages that the coalition submitted as queries to the $\mathsf{Sign}_\zeta^{\mathcal{O}}(\cdot, t)$ oracle. Note that $\mathcal{Q}_t$ does not contain messages submitted as queries to $\mathsf{Sign}_\ell^{\mathcal{O}}(\cdot, t)$ for $\ell \neq \zeta$.
3. The coalition C outputs a signature triple (x, t, σ).
4. The output of the game is defined to be 1 if and only if one of the following conditions is met:
   a. $U_v \notin C$ and σ is a (ζ, v)-fraudulent signature on x for period t; or
   b. $U_\zeta \notin C$ and σ is a ζ-authentic signature on x for period t, where $x \notin \mathcal{Q}_t$ and $t \notin T'$.

Definition 8.7. Let Π be a KI-USS scheme (with notation as defined in Definition 8.1) with security parameter k and let $\epsilon(k)$ be a negligible function of k. We say Π is *strongly* $(\omega, \gamma, \psi_S, \psi_V, \epsilon)$*-unforgeable* if for all coalitions C of at most ω users, and all choices of target signer U_ζ and verifier U_v, it holds that

$$\Pr[\text{KI-Sig-forge}_{C,\Pi}(k) = 1] \leq \epsilon(k).$$

Given the nature of signing key exposure, it is reasonable to consider a scenario in which two or more *consecutive* time periods are compromised by the adversary. In this case, it is quite possible (or even likely) that the adversary gains access not only to the signing keys from the exposed periods, but also the key-updating information sent from the user's secure device between those compromised time periods. To protect against this, it is useful to consider the notion of *secure key updates* [5], which says that the combination of signing information from two consecutive exposed periods $t - 1$ and t, together with the key-updating information between these periods, should be equivalent to the signing information from these two periods alone.

Definition 8.8. Let Π be a strongly $(\omega, \gamma, \psi_S, \psi_V, \epsilon)$-unforgeable KI-USS scheme. Suppose a coalition C of at most ω users plays the signature game of Definition 8.6, with the modification that any time C accesses, for a user $U_\ell \notin C$ and two consecutive time periods $t - 1$ and t, the oracles SigningExposure$^{\mathcal{O}}(U_\ell, t - 1)$ and SigningExposure$^{\mathcal{O}}(U_\ell, t)$, the coalition C receives the additional information $mk_\ell^{(t-1,t)}$, together with U_ℓ's signing information from periods $t - 1$ and t. If Π satisfies Definition 8.7 with this new signature game, we say that Π has *secure key updates*.

9 Construction: USS scheme with key insulation

We now give an extension to the USS construction from Hanaoka et al. [7], which we analyzed in Section 7. The extension presented here uses ideas from a multi-receiver authentication code construction presented by Seito et al. [14].

The construction given here is very similar to the basic construction given in Section 7, except that we need our polynomial construction to be divided into two pieces, so that we can split each user's signing algorithm into an initial signing key which is stored on the user's insecure device and a master signing key which is stored on the user's secure device.

To that end, we use a polynomial F of the same form as the basic construction and a polynomial mk, which has a similar form as F but is extended to take into account time periods. The polynomials F and mk are used to construct (by substituting a user's identity into these polynomials, as before) each user's initial signing key and master signing key, respectively. A user's overall signing information for a particular time period is the sum of these two polynomials evaluated at the user's identity and the given time period.

9.1 Key pair generation

Let q be a prime power such that $q > n$. (In practice, we pick q to be much larger than n.) Let $\mathbb{F}_q$ be a finite field with q elements. The TI picks n *verification vectors* $\vec{v_1}, \ldots, \vec{v_n} \in (\mathbb{F}_q)^\omega$ uniformly at random for users $U_1, \ldots, U_n$, respectively, subject to one additional constraint. For technical reasons, we assume the verification vectors $\vec{v_1}, \ldots, \vec{v_n} \in (\mathbb{F}_q)^\omega$ satisfy the additional property that for any subset of size $\omega + 1$, the corresponding subset of size $\omega + 1$ formed from the new vectors $[1, \vec{v_1}], \ldots, [1, \vec{v_n}] \in (\mathbb{F}_q)^{\omega+1}$ is a linearly independent set. We assume user identities $U_1, \ldots, U_n$ and time periods $\{1, \ldots, N\}$ have a representation as elements in $\mathbb{F}_q$ in some suitable (and public) way.

The TI constructs two polynomials:

1. The polynomial $F(x, y_1, \ldots, y_\omega, z)$, where

$$F(x, y_1, \ldots, y_\omega, z) = \sum_{i=0}^{n-1} \sum_{k=0}^{\psi} a_{i0k0} x^i z^k + \sum_{i=0}^{n-1} \sum_{j=1}^{\omega} \sum_{k=0}^{\psi} a_{ijk0} x^i y_j z^k,$$

   where the coefficients $a_{ijk0} \in \mathbb{F}_q$ are chosen uniformly at random.

2. The polynomial $mk(x, y_1, \ldots, y_\omega, z, t)$, where

$$mk(x, y_1, \ldots, y_\omega, z, t) = \sum_{i=0}^{n-1} \sum_{k=0}^{\psi} \sum_{\ell=1}^{\gamma} a_{i0k\ell} x^i z^k t^\ell + \sum_{i=0}^{n-1} \sum_{j=1}^{\omega} \sum_{k=0}^{\psi} \sum_{\ell=1}^{\gamma} a_{ijk\ell} x^i y_j z^k t^\ell.$$

For each user U_ζ for $1 \le \zeta \le n$, the TI computes the *initial signing key*

$$sk_\zeta^0(y_1, \ldots, y_\omega, z) = F(x, y_1, \ldots, y_\omega, z)|_{x=U_\zeta},$$

the *master signing key*

$$mk_\zeta(y_1, \ldots, y_\omega, z, t) = mk(x, y_1, \ldots, y_\omega, z, t)|_{x=U_\zeta},$$

and the *verification key*

$$\tilde{v}_\zeta(x, z, t) = F(x, y_1, \ldots, y_\omega, z)|_{(y_1,\ldots,y_\omega)=\vec{v_\zeta}} + mk(x, y_1, \ldots, y_\omega, z, t)|_{(y_1,\ldots,y_\omega)=\vec{v_\zeta}}.$$

It is assumed the TI sends sk_ζ^0, mk_ζ, $\vec{v_\zeta}$, and $\tilde{v}_\zeta$ to the corresponding user via a secure channel and deletes the information from his memory afterwards. The user U_ζ places his master signing key $mk_\zeta(y_1, \ldots, y_\omega, z, t)$ on his secure device H_ζ and deletes this information from his memory.

9.2 Updating phase

To update his signing key from a time period t_0 to the next time period t_1, a user U_ζ requests key-updating information from the secure device H_ζ. The device H_ζ computes

$$mk_\zeta^{(t_0,t_1)}(y_1,\ldots,y_\omega,z) := mk_\zeta(y_1,\ldots,y_\omega,z,t)|_{t=t_1} - mk_\zeta(y_1,\ldots,y_\omega,z,t)|_{t=t_0}$$

and sends this polynomial via a secure channel to U_ζ.

The user U_ζ then computes

$$\mathsf{Sign}_\zeta^{(t_1)}(y_1,\ldots,y_\omega,z) = \mathsf{Sign}_\zeta^{(t_0)}(y_1,\ldots,y_\omega,z) + mk_\zeta^{(t_0,t_1)}(y_1,\ldots,y_\omega,z),$$

where the signing key for time period $t = 1$ is defined by

$$\mathsf{Sign}_\zeta^{(1)} = sk_\zeta^0(y_1,\ldots,y_\omega,z) + mk_\zeta^{(0,1)}(x,y_1,\ldots,y_\omega,z).$$

Remark 9.1. For a given time period $t_0 > 0$, user U_ζ's signing key is as follows:

$$\mathsf{Sign}_\zeta^{(t_0)}(y_1,\ldots,y_\omega,z) = F(x,y_1,\ldots,y_\omega,z)|_{x=U_\zeta} + mk(x,y_1,\ldots,y_\omega,z,t)|_{x=U_\zeta,t=t_0}.$$

9.3 Signature generation and verification

For a message $m \in \mathbb{F}_q$ during time period t_0, U_ζ generates a signature by

$$\sigma(y_1,\ldots,y_\omega) = \mathsf{Sign}_\zeta^{(t_0)}(y_1,\ldots,y_\omega,z)|_{z=m}.$$

To verify a signature pair (t_0,σ) from U_ζ on a message m, a user U_ν checks that

$$\sigma(y_1,\ldots,y_\omega)|_{(y_1,\ldots,y_\omega)=\vec{v}_\nu} = \tilde{v}_\nu(x,z,t)|_{x=U_\zeta,z=m,t=t_0}.$$

Remark 9.2. As in the basic construction, the parameter ω determines the maximum number of colluders the scheme protects against and the parameter ψ determines the maximum number of signatures (on unique messages) each user can produce without revealing their signing information. Similarly, the parameter γ is the maximum number of time periods for which a user U_h's temporary signing key can be compromised (so long as U_h's master signing key is not exposed).

9.4 Security analysis

We consider the security of this construction in a restricted model, specified as follows. We let $\mathcal{Q}$ denote the set of messages that the coalition submitted as queries to the $\mathsf{Sign}_\zeta^{\mathcal{O}}$ oracle. We then replace Condition 4b of Definition 8.6 with the following:

4b. $U_\zeta \notin C$ and σ is a ζ-authentic signature on x for period t, where $x \notin \mathcal{Q}$ and $t \notin T'$.

This weakened condition allows forgeries (x,t,σ) for signer U_ζ in the case where a ζ-authentic signature for some time $t' \neq t$ is known (and U_ζ's signing key for time period t has not been exposed). We can mitigate the impact of this type of forgery in our construction by assuming messages m contain effective dates for signatures. In this sense, an adversary can create a new signature on m for a different time period once he has seen the first signature on m, but the effective date of the signature will remain the same, i.e., this type of forgery will be detectable.

Given q, we define the security parameter to be k, where $k = \log_2 q$. We consider KI-Sig-forge$_{C,\Pi}(k)$ and calculate the probability that the output is 1. In particular, we consider the probability that the coalition C produces a signature triple (m,t',σ) satisfying Conditions 4a and 4b of Definition 8.6 separately. Here C is allowed, for each $U_h \notin C$, either γ queries to SigningExposure $\mathcal{O}(U_h,\cdot)$ or the single query MasterExposure $\mathcal{O}(U_h)$,

but not both. We prove the scheme is unforgeable with respect to coalitions C of size at most ω, where C is allowed $\psi_S = (n-\omega)\psi$ oracle queries to $\mathsf{Sign}^{\mathcal{O}}$ (where ψ is the total number of $\mathsf{Sign}_h^{\mathcal{O}}$ oracle queries allowed for each user $U_h \notin C$), and where the number of $\mathsf{Vrfy}^{\mathcal{O}}$ queries, say ψ_V, is arbitrary. (As shown in the following theorem, the probability that C creates a successful forgery depends on this value ψ_V.) That is, we allow C to have at most ω members and to have access to ψ sample signatures from each user $U_h \notin C$. (This is consistent with the fact that in this USS scheme, each user is allowed to produce at most ψ signatures, so the bound on oracle access to $\mathsf{Sign}_h^{\mathcal{O}}$ for each user $U_h \notin C$ must be ψ.)

Theorem 9.3. *Under the above assumptions, the coalition C outputs a signature triple (m, t', σ) in the game* KI-Sig-forge$_{C,\Pi}(k)$ *of Definition 8.6 satisfying Condition 4a or 4b with probability at most $\frac{r}{1-r}$, where $r = \frac{\psi_V}{q-1}$.*

Proof. Recall the assumption that the verification vectors $\vec{v_1}, \ldots, \vec{v_n} \in (\mathbb{F}_q)^\omega$ satisfy the additional property that for any subset of size $\omega+1$, the corresponding subset of size $\omega+1$ formed from the new vectors $[1, \vec{v_1}], \ldots, [1, \vec{v_n}] \in (\mathbb{F}_q)^{\omega+1}$ is a linearly independent set. We use this fact throughout the proof.

We consider the strongest possible coalition C. To this end, we consider a coalition of size ω whose verification vectors form a linearly independent set. Lemma A.1 implies that this is always possible. Without loss of generality, assume our adversaries are $C = \{U_1, \ldots, U_\omega\}$, with target signer U_ζ and target verifier U_ν. We assume that the coalition C outputs a signature (t', σ) for some choice of time period t' and message m, with claimed signer U_ζ.

For ease of notation, define $y_0 = 1$ and let $\vec{y}$ denote the vector $(y_0, y_1, \ldots, y_\omega)$. Let $G(x, \vec{y}, z, t)$ denote $F(x, \vec{y}, z) + mk(x, \vec{y}, z, t)$. Then

$$G(x, \vec{y}, z, t) = \sum_{i=0}^{n-1} \sum_{j=0}^{\omega} \sum_{k=0}^{\psi} \sum_{\ell=0}^{\gamma} a_{ijk\ell} x^i y_j z^k t^\ell.$$

We sometimes refer to a user U_h's augmented verification vector, namely $[1, \vec{v_h}]$, as $(v_{h,0}, \ldots, v_{h,\omega})$.

The polynomial F is determined by the $n(\omega+1)(\psi+1)(\gamma+1)$ unknown coefficients $a_{ijk\ell}$. The coalition C has access to the following information:

1. The verification keys $\tilde{v}_1, \ldots, \tilde{v}_\omega$. We have, for $U_h \in C$,

$$\tilde{v}_h(x, z, t) = G(x, \vec{y}, z, t)|_{\vec{y}=\vec{v_h}} = \sum_{i=0}^{n-1} \sum_{j=0}^{\omega} \sum_{k=0}^{\psi} \sum_{\ell=0}^{\gamma} v_{h,j} a_{ijk\ell} x^i z^k t^\ell.$$

   Noting that $\tilde{v}_h$ is a polynomial with terms of the form $(c_{ik\ell})_h x^i z^k t^\ell$ for $0 \le i \le n-1$, $0 \le k \le \psi$, and $0 \le \ell \le \gamma$, we have that C has access to $n(\psi+1)(\gamma+1)$ equations $C_{ik\ell h}$ in the unknown coefficients $a_{ijk\ell}$ for each $U_h \in C$, where

$$C_{ik\ell h} : \sum_{j=0}^{\omega} v_{h,j} a_{ijk\ell} = (c_{ik\ell})_h$$

   for some (known) $(c_{ik\ell})_h \in \mathbb{F}_q$. We note these equations

$$\{C_{ik\ell h} : 0 \le i \le n-1,\ 0 \le k \le \psi,\ 0 \le \ell \le \gamma,\ 1 \le h \le \omega\} \tag{9.1}$$

   form a linearly independent set, since the rank of $\{(1, \vec{v_1}), \ldots, (1, \vec{v_\omega})\} \subseteq (\mathbb{F}_q)^{\omega+1}$ is ω.
2. The signing information for each $U_h \in C$. That is, the initial signing keys $sk_h^0(\vec{y}, z) = F(U_h, \vec{y}, z)$, as well as the master signing keys $mk_h(\vec{y}, z, t) = mk(U_h, \vec{y}, z, t)$.
   Rewriting these equations, we have

$$sk_h^0(\vec{y}, z) = \sum_{i=0}^{n-1} \sum_{j=0}^{\omega} \sum_{k=0}^{\psi} U_h^i a_{ijk0} y_j z^k$$

   and

$$mk_h(\vec{y}, z, t) = \sum_{i=0}^{n-1} \sum_{j=0}^{\omega} \sum_{k=0}^{\psi} \sum_{\ell=1}^{\gamma} U_h^i a_{ijk\ell} y_j z^k t^\ell.$$

We observe that both sk_h^0 and mk_h are polynomials with terms of the form $(d_{jk\ell})_h y_j z^k t^\ell$, for $0 \le j \le \omega$, $0 \le k \le \psi$, and $0 \le \ell \le \gamma$. So C has access to $(\omega+1)(\psi+1)(\gamma+1)$ equations $D_{jk\ell h}$ in the unknown coefficients $a_{ijk\ell}$ for each $U_h \in C$, where

$$D_{jk\ell h}\colon \sum_{i=0}^{n-1} U_h^i a_{ijk\ell} = (d_{jk\ell})_h$$

for some (known) $(d_{jk\ell})_h \in \mathbb{F}_q$. Now, these equations, together with the equations from (9.1), are not a linearly independent set, due to the relationships between users' signing and verification keys. More specifically, for any users U_h and U_h', we have

$$sk_h^0(\vec{y}, z)|_{\vec{y}=\vec{v_{h'}}} + mk_h(\vec{y}, z, t)|_{\vec{y}=\vec{v_{h'}}} = \tilde{v}_{h'}(x, z, t)|_{x=U_h}. \tag{9.2}$$

Equation (9.2) implies that for each $U_h \in C$ and each choice of k and ℓ, for $0 \le k \le \psi$ and $0 \le \ell \le \gamma$, we have a set of ω relations among the $\omega+1$ equations $\{D_{jk\ell h} : 0 \le j \le \omega\}$.
Thus, the information gleaned from the coalition's signing information is contained in the set

$$\{D_{0k\ell h} : 0 \le k \le \psi,\ 0 \le \ell \le \gamma,\ 1 \le h \le \omega\}. \tag{9.3}$$

3. Key exposure information for honest users. For each $U_h \notin C$, we allow the coalition either signing key exposure or master key exposure (but not both for a given user). For a given $U_h \notin C$, this information takes one of the following forms.
   - Signing key exposure for $U_h \notin C$: C has access to $\mathsf{Sign}_h^{t_{h_1}}, \ldots, \mathsf{Sign}_h^{t_{h_\gamma}}$, where $t_{h_1}, \ldots, t_{h_\gamma}$ are valid time periods. We have, for a given time period t_{h_d} (where $1 \le d \le \gamma$),

$$\mathsf{Sign}_h^{t_{h_d}}(\vec{y}, z) = G(U_h, \vec{y}, z, t_{h_d}) = \sum_{i=0}^{n-1}\sum_{j=0}^{\omega}\sum_{k=0}^{\psi}\sum_{\ell=0}^{\gamma} a_{ijk\ell} U_h^i (t_{h_d})^\ell y_j z^k.$$

     Note that $\mathsf{Sign}_h^{t_{h_d}}$ is a polynomial with terms of the form $(e_{jk})^{t_{h_d}} y_j z^k$ for $0 \le j \le \omega$ and $0 \le k \le \psi$, so the coalition C has access to equations $E_{jkt_{h_d}}$ in the unknown coefficients $a_{ijk\ell}$, where

$$E_{jkt_{h_d}}\colon \sum_{i=0}^{n-1}\sum_{\ell=0}^{\gamma} a_{ijk\ell} U_h^i (t_{h_d})^\ell = (e_{jk})^{t_{h_d}}$$

     for some (known) $(e_{jk})^{t_{h_d}} \in \mathbb{F}_q$. In a manner similar to the previous analysis, we observe the relation

$$\mathsf{Sign}_h^{t_{h_d}}(\vec{y}, z)|_{\vec{y}=\vec{v_{h'}}} = \tilde{v}_{h'}(x, z, t)|_{x=U_h, t=t_{h_d}}$$

     for any pair of users U_h and $U_{h'}$. Thus, considering $U_{h'} \in C$ and fixing k, we have a set of ω relations among the $\omega+1$ equations $\{E_{jkt_{h_d}} : 0 \le j \le \omega\}$. This implies that any new information gained by signing key exposure for the user U_h is contained in the set

$$\{E_{0kt_{h_d}} : 0 \le k \le \psi,\ 1 \le d \le \gamma\}. \tag{9.4}$$

   - Master key exposure for $U_h \notin C$:

$$mk_h(\vec{y}, z, t) = \sum_{i=0}^{n-1}\sum_{j=0}^{\omega}\sum_{k=0}^{\psi}\sum_{\ell=1}^{\gamma} U_h^i a_{ijk\ell} y_j z^k t^\ell.$$

     Now, mk_h is a polynomial with terms of the form $(d_{jk\ell})_e y_j z^k t^\ell$, for $0 \le j \le \omega$, $0 \le k \le \psi$, and $1 \le \ell \le \gamma$. So C has access to $(\omega+1)(\psi+1)(\gamma)$ equations $D_{jk\ell h}$ in the unknown coefficients $a_{ijk\ell}$ for each $U_h \notin C$, where

$$D_{jk\ell h}\colon \sum_{i=0}^{n-1} U_h^i a_{ijk} = (d_{jk\ell})_h$$

for some (known) $(d_{jk\ell})_h \in \mathbb{F}_q$. As before, the relation between users' signing and verification keys implies that it suffices to consider the set

$$\{D_{0k\ell h} : 0 \le k \le \psi,\ 1 \le \ell \le \gamma\} \tag{9.5}$$

for the user U_h.

4. Signing oracle queries for each user $U_h \notin C$. The coalition C has access to ψ signing oracle queries for each user $U_h \notin C$. We first observe that for both types of key exposure, the result of a signing oracle query on message m for a period t_0 contains enough information for C to determine the signature on m for all other time periods. This is easy to see for the case of master key exposure, since C can compute $F(x, y, z)|_{x=U_h, z=m}$ by subtracting $mk_h(\vec{y}, z, t)|_{z=m, t=t_0}$ from the signature. This is the case for signing key exposure so long as the coalition C maximizes its information by requesting a signature for a time period for which C does not already have the signing key. In this case, C knows a signature on m in $\gamma + 1$ time periods, so he can solve for $G(x, \vec{y}, z, t)|_{x=U_h, z=m}$, as this is a polynomial of degree γ in t. Therefore the time period requested in a signature oracle query is irrelevant to this analysis; for simplicity we use t_h as a placeholder for the time period in requested signatures from signer U_h.
That is, C has access to up to ψ signatures $\sigma_{h,k'}$ from each user $U_h \notin C$, on messages $m_{h,k'}$ of C's choice, where $1 \le k' \le \psi$, with the exception that C can only access a signature $\sigma_{\zeta,k'}$ on a message $m_{\zeta,k'} \ne m$ with signer U_ζ.
Each requested signature has the form

$$\sigma_{h,k'} = G(x, \vec{y}, z, t)|_{x=U_h, z=m_{h,k'}, t=t_h} = \sum_{i=0}^{n-1} \sum_{j=0}^{\omega} \sum_{k=0}^{\psi} \sum_{\ell=0}^{\gamma} a_{ijk\ell} U_h^i (m_{h,k'})^k (t_h)^\ell y_j.$$

Note that $\sigma_{h,k'}$ is a polynomial with terms of the form $(b_j)_{h,k'} y_j$ for $0 \le j \le \omega$, so C has access to equations $B_{jhk'}$, where

$$B_{jhk'} : \sum_{i=0}^{n-1} \sum_{k=0}^{\psi} \sum_{\ell=0}^{\gamma} a_{ijk\ell} U_h^i (m_{h,k'})^k (t_h)^\ell = (b_j)_{h,k'}$$

for some (known) $(b_j)_{h,k'} \in \mathbb{F}_q$. As before, we have

$$\sigma_{h,k'}^{t_h}(\vec{y})|_{\vec{y}=\vec{v_{h'}}} = \tilde{v}_{h'}(x, z, t)|_{x=U_h, z=m_{h,k'}, t=t_h}$$

for each $U_{h'} \in C$. Thus it suffices to consider the set

$$\{B_{0hk'} : 1 \le k' \le \psi\} \tag{9.6}$$

for each $U_h \notin C$.

5. Up to ψ_V query results from the oracle $\mathsf{Vrfy}_h^{\mathcal{O}}$ for $U_h \notin C$. In the following, we discuss the attack scenario without $\mathsf{Vrfy}^{\mathcal{O}}$ queries. Incorporating these queries into the analysis follows as in the proof of Theorem 7.2.

To summarize, the information obtained by the coalition C is contained in the following sets of equations: sets (9.1) and (9.3), together with, for each $U_h \notin C$, one of set (9.4) or set (9.5) (depending on the type of key exposure), and set (9.6). These equations do form a linearly independent set; we provide the proof in Appendix A.2. We have a total of $n\omega(\psi + 1)(\gamma + 1) + \omega(\psi + 1)(\gamma + 1) + (n - \omega)\gamma(\psi + 1) + (n - \omega)\psi$ equations, which implies that we have $n - \omega$ free variables in the given linear system.

With the given information, C can consider the polynomials $G'(x, \vec{y}, z, t)$ consistent with the known information about $G(x, \vec{y}, z, t)$. If a given polynomial G' is consistent with the known information about G, we say G' satisfies property $(*)$. We let

$$\mathcal{G} = \{G'(x, \vec{y}, z, t) : G' \text{ satisfies } (*)\}.$$

From above, we have $|\mathcal{G}| = q^{n-\omega}$.

Case 1: $U_\zeta \notin C$, $U_\nu \in C$. In this case, the goal of C is to produce a ζ-authentic signature (m, σ) for some time period t' (for which C does not already have the corresponding signing key). We first observe that producing a ζ-authentic signature for such a time period t' is equivalent to producing U_ζ's general signing key $\mathsf{Sign}_\zeta(\vec{y}, z, t) = G(x, \vec{y}, z, t)|_{x=U_\zeta}$. Once we have this result, the rest of the proof for this case is almost identical to that provided in Theorem 7.2, so we do not provide the details here.

To see this, suppose C produces a ζ-authentic signature (m, σ) for time period $t' \neq t_{\zeta_i}$ for $1 \leq i \leq \gamma$ consistent with C's information. Then C has access to a total of $\psi + 1$ points (namely, the signatures on m and on $m_{\zeta,1}, \ldots, m_{\zeta,\psi}$) on $G(x, \vec{y}, z, t)|_{x=U_\zeta, t=t'}$, which is a polynomial of degree ψ in z. Thus C can solve for

$$\mathsf{Sign}_\zeta^{t'}(\vec{y}, z) = G(x, \vec{y}, z, t)|_{x=U_\zeta, t=t'},$$

so C knows U_ζ's signing key for the time period t'. There are then two cases to consider, depending on which type of key exposure C has for the target signer U_ζ:

1. Suppose C has signing key exposure against U_ζ for γ time periods, denoted by $t_{\zeta_1}, \ldots, t_{\zeta_\gamma}$. Then C knows $\mathsf{Sign}_\zeta^{t'}, \mathsf{Sign}_\zeta^{t_{\zeta_1}}, \ldots, \mathsf{Sign}_\zeta^{t_{\zeta_\gamma}}$, i.e., $\gamma + 1$ points on $\mathsf{Sign}_\zeta(\vec{y}, z, t)$, which is a polynomial of degree γ in t. So C can solve for $\mathsf{Sign}_\zeta(\vec{y}, z, t)$.
2. Suppose C has master key exposure for U_ζ, so C knows $mk_\zeta(\vec{y}, z, t)$. Then

$$\mathsf{Sign}_\zeta^{t'}(\vec{y}, z) - mk_\zeta(\vec{y}, z, t)|_{t=t'} = (G - mk)(x, \vec{y}, z, t)|_{x=U_\zeta, t=t'} = F(x, \vec{y}, z)|_{x=U_\zeta}.$$

That is, C knows both $F(x, \vec{y}, z)|_{x=U_\zeta}$ and $mk_\zeta(\vec{y}, z, t)$, the sum of which yields $\mathsf{Sign}_\zeta(\vec{y}, z, t)$.

Case 2: $U_\zeta \notin C$, $U_\nu \notin C$ *and Case 3:* $U_\zeta \in C$, $U_\nu \notin C$. The case where $U_\zeta \notin C$, $U_\nu \notin C$ and the case where $U_\zeta \in C$, $U_\nu \notin C$ follow the same argument as for the basic USS scheme provided in the proof of Theorem 7.2, so we do not reproduce the proof here. □

Theorem 9.4. *The above scheme has secure key updates.*

Proof. This is easy to see from the scheme definition. For a given user U_h, consider the signing information $\mathsf{Sign}_h^{t_{h_1}}$ and $\mathsf{Sign}_h^{t_{h_2}}$ from the two consecutive periods t_{h_1} and t_{h_2}. We see that

$$\mathsf{Sign}_h^{t_{h_2}} - \mathsf{Sign}_h^{t_{h_1}} = mk_h^{(t_{h_1}, t_{h_2})},$$

which is the key-updating information from period t_{h_1} to t_{h_2}. □

10 Conclusion

We have presented a new security model for unconditionally secure signature schemes, one which fully treats the implications of having multiple verification algorithms. In particular, we have given a formal discussion of dispute resolution, a necessary component of any USS scheme, and analyzed the effect of dispute resolution on unforgeability. We have provided formal definitions of non-repudiation and transferability, and given sufficient conditions for a USS scheme to satisfy these properties. Moreover, we have analyzed the trust assumptions required in typical examples of dispute resolution. We have given an analysis of Hanaoka et al.'s construction [7] in our security model. Finally, we have provided an extension of our basic framework to the setting of key-insulation and presented a construction, inspired by the original construction of Hanaoka et al. [7] and the work of Seito et al. [14] and Seito and Shikata [15], which satisfies our security definitions.

A Analysis of constructions

We need the following lemmas:

Lemma A.1. *Let $n \in \mathbb{N}$ and consider the set of $n+1$ vectors*

$$R = \{\vec{r}_i = (r_{i,1}, \ldots, r_{i,n}) \in (\mathbb{F}_q)^n : i = 1, \ldots, n+1\}.$$

If the vectors $\{\vec{r}_i{}' = (1, r_{i,1}, \ldots, r_{i,n}) \in (\mathbb{F}_q)^{n+1} : i = 1, \ldots, n+1\}$ form a linearly independent set, then there exists a subset $R' \subset R$ of linearly independent vectors of size n.

Proof. Consider the matrix

$$M = \begin{pmatrix} 1 & r_{1,1} & r_{1,2} & \ldots & r_{1,n} \\ 1 & r_{2,1} & r_{2,2} & \ldots & r_{2,n} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & r_{n+1,1} & r_{n+1,2} & & r_{n+1,n} \end{pmatrix}.$$

Let M_{ij} denote the (i, j) minor matrix of M. Then calculating the determinant of M by expansion along the first column, we have

$$\det(M) = \sum_{i=1}^{n+1} (-1)^{i+1} \det(M_{i,1}). \tag{A.1}$$

Recall M is invertible, so $\det(M) \neq 0$. Thus (A.1) implies $\det(M_{k,1}) \neq 0$ for some $k \in \{1, \ldots, n\}$. We conclude that the matrix $M_{k,1}$ is invertible, so the desired subset R' exists. □

Lemma A.2. *Let $n \in \mathbb{N}$ and let F and F' be polynomials in $y_1, \ldots, y_n$ of the form $a_0 + \sum_{i=1}^{n} a_i y_i$ over $\mathbb{F}_q$. Suppose F' and F agree on the $n+1$ vectors*

$$R = \{\vec{r}_i = (r_{i,1}, \ldots, r_{i,n}) \in (\mathbb{F}_q)^n : i = 1, \ldots, n+1\}.$$

If the vectors $\{\vec{r}_i{}' = (1, r_{i,1}, \ldots, r_{i,n}) \in (\mathbb{F}_q)^{n+1} : i = 1, \ldots, n+1\}$ form a linearly independent set, then $F' = F$.

Proof. Define linear homogeneous polynomials $G, G' \in \mathbb{F}_q[y_0, \ldots, y_n]$ such that

$$G(y_0, \ldots, y_n)|_{y_0=1} = F(y_1, \ldots, y_n)$$

and

$$G'(y_0, \ldots, y_n)|_{y_0=1} = F'(y_1, \ldots, y_n).$$

We have $(G - G')(1, r_{i,1}, \ldots, r_{i,n}) = 0$ for $1 \leq i \leq n+1$. In particular, this forms a homogeneous linear system of $n+1$ equations in the $n+1$ unknowns $a_0, \ldots, a_n$. Since the vectors $(1, r_{i,1}, \ldots, r_{i,n}) \in (\mathbb{F}_q)^{n+1}$ for $i = 1, \ldots, n+1$ are linearly independent, it follows that $G - G'$ is the zero polynomial, so $G = G'$. Hence $F = F'$, as desired. □

A.1 Basic construction: Proof of linear independence

We use assumptions and notation as in the proof of Theorem 7.2. Recall that the information obtained by the coalition C is contained in equation sets (7.1) and (7.3), together with, for each $U_h \notin C$, equation set (7.4). We have a total of $n\omega\psi + n\omega + \omega + \psi n$ equations, which would imply there are at least $n - \omega$ free variables in the given linear system.

We proceed by showing that allowing C access to an additional $n - \omega$ equations (in the form of sample signatures from each user not in C) suffices to solve the linear system. This implies the linear independence of the original set of equations, as desired.

Lemma A.3. *Let $U_h \notin C$. Suppose C has access to an additional h-authentic signature from U_h on some message $m_{h,\psi+1}$ satisfying $m_{h,\psi+1} \neq m_{h,k}$ for $1 \leq k \leq \psi$. Then this is equivalent to C having access to all of the signing information from U_h.*

Proof. This follows immediately from the fact that U_h's signing algorithm $s_h(\vec{y}, z)$ is a polynomial of degree $\psi + 1$ in z. □

Lemma A.3 implies that the system of equations

$$\{C_{ikh} : 0 \le i \le n-1,\ 0 \le k \le \psi,\ 1 \le h \le \omega\} \cup \{D_{0kh} : 0 \le k \le \psi,\ 1 \le h \le n\}$$

is equivalent to the original system of equations known to C, plus $n - \omega$ additional equations

$$\{B_{0h(\psi+1)} : U_h \notin C\}$$

(obtained from an extra h-authentic signature on some new message $m_{h,\psi+1}$ for each $U_h \notin C$). In the following lemma, we show this new set is linearly independent, and therefore the linear independence of the original set follows.

Lemma A.4. *The coefficient matrix formed from the equations*

$$\{C_{ikh} : 0 \le i \le n-1,\ 0 \le k \le \psi,\ 1 \le h \le \omega\} \cup \{D_{0kh} : 0 \le k \le \psi,\ 1 \le h \le n\}$$

has nonzero determinant.

Proof. The coefficient matrix E is a block matrix of the form

$$E = \left[\begin{array}{c:c} A & 0 \\ \hdashline C & D \end{array}\right],$$

where A and D are square matrices. Thus the determinant of the coefficient matrix, $\det(E)$, is defined by $\det(E) = \det(A)\det(D)$. We show that $\det(E) \neq 0$.

Here the submatrix

$$\left[\begin{array}{c:c} A & 0 \end{array}\right]$$

is derived from the equations $\{D_{0kh} : 0 \le k \le \psi, 1 \le h \le n\}$, where

$$A = \begin{bmatrix} V_n & 0 & \cdots & 0 \\ 0 & V_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & V_n \end{bmatrix}$$

is a diagonal matrix with $(\psi + 1)$ Vandermonde matrices V_n on the diagonal. That is, we have

$$V_n = \begin{bmatrix} 1 & U_1 & \cdots & U_1^{n-1} \\ 1 & U_2 & \cdots & U_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & U_n & \cdots & U_n^{n-1} \end{bmatrix}.$$

To see that A is invertible, note that $\det(A) = \det(V_n)^{\psi+1} \neq 0$.

The submatrix

$$\left[\begin{array}{c:c} C & D \end{array}\right]$$

is derived from the equations $\{C_{ikh} : 0 \le i \le n-1, 0 \le k \le \psi, 1 \le h \le \omega\}$.

The matrix D is defined by

$$D = \begin{bmatrix} v_{1,1}\mathrm{I} & v_{1,2}\mathrm{I} & \cdots & v_{1,\omega}\mathrm{I} \\ v_{2,1}\mathrm{I} & v_{2,2}\mathrm{I} & \cdots & v_{2,\omega}\mathrm{I} \\ \vdots & \vdots & \ddots & \vdots \\ v_{\omega,1}\mathrm{I} & v_{\omega,2}\mathrm{I} & \cdots & v_{\omega,\omega}\mathrm{I} \end{bmatrix},$$

where I is the $n(\psi + 1) \times n(\psi + 1)$ identity matrix.

The fact that $\det(D) \neq 0$ follows immediately from the linear independence of the coalition's verification keys $\{\vec{v_h} : 1 \le h \le \omega\}$. □

Remark A.5. The security analysis for a general coalition (whose verification keys may or may not be linearly independent) is very similar. Recall the assumption that the n elements $\vec{v}_1, \ldots, \vec{v}_n \in (\mathbb{F}_q)^\omega$ satisfy the additional property that for any subset of size $\omega + 1$, the corresponding subset of size $\omega + 1$ formed from the new vectors $[1, \vec{v}_1], \ldots, [1, \vec{v}_n] \in (\mathbb{F}_q)^{\omega+1}$ is a linearly independent set. Consider a possible coalition C of size ω, where $V = \{\vec{v}_h : U_h \in C\}$ is the set of C's verification keys. Then Lemma A.1 implies that, for any $\vec{v}_r \notin V$, we can pick a subset of size ω with full rank from $V \cup \{\vec{v}_r\}$.

There are then two cases. Either the set V has rank ω, so that V forms a basis for $(\mathbb{F}_q)^\omega$, or the additional vector $\vec{v}_r$ is needed to form a basis. In the former case, the analysis is as above. In the latter, the span of V is a subspace of $(\mathbb{F}_q)^\omega$ of dimension $\omega - 1$. The linear system corresponding to C's information has $n - (\omega - 1)$ free variables, which can be shown using a linear algebra trick similar to the one used above.

In fact, if we did not have any additional assumptions on user verification keys (other than that they are chosen uniformly at random from $(\mathbb{F}_q)^\omega$), the proof follows much as before. A coalition C's information in this case depends on the rank of V, i.e., the linear system has $n - r$ free variables, where $r = \text{rank}(V)$.

A.2 Key insulation construction: Proof of linear independence

We use assumptions and notation as in the proof of Theorem 9.3. Recall that the information obtained by the coalition C is contained in the following sets of equations: sets (9.1) and (9.3), together with, for each $U_h \notin C$, one of set (9.4) or set (9.5) (depending on the type of key exposure), and set (9.6). We have a total of $n\omega(\psi + 1)(\gamma + 1) + \omega(\psi + 1)(\gamma + 1) + (n - \omega)\gamma(\psi + 1) + (n - \omega)\psi$ equations, which implies that we have $n - \omega$ free variables in the given linear system.

We use the same method as in Appendix A.1; we include the argument here for completeness. We proceed by showing that allowing C access to an additional $n - \omega$ equations (in the form of sample signatures from each user not in C) suffices to solve the linear system. This implies the linear independence of the original set of equations, as desired.

Lemma A.6. *Let $U_h \notin C$. Suppose C has access to an additional h-authentic signature from U_h on some message $m_{h,\psi+1}$ satisfying $m_{h,\psi+1} \neq m_{h,k}$ for $1 \leq k \leq \psi$ in addition to either master key or signing key exposure from U_h. Then this is equivalent to C having access to all of the signing information from U_h.*

Proof. Consider a user $U_h \notin C$. Then C has access to up to ψ sample signatures from U_h on distinct messages $m_{h,k}$ for $1 \leq k \leq \psi$, which yield the equations $\{B_{0hk'} : 1 \leq k' \leq \psi\}$. Suppose C has access to one additional signature from U_h, yielding the additional equation $B_{0h(\psi+1)}$.

Now suppose C has achieved master key exposure for U_h. Then the coalition C has access to the set $\{D_{0k\ell h} : 0 \leq k \leq \psi, 1 \leq \ell \leq \gamma\}$. Let

$$B_h = \{D_{0k\ell h} : 0 \leq k \leq \psi,\ 1 \leq \ell \leq \gamma\} \cup \{B_{0hk'} : 1 \leq k' \leq \psi\}.$$

We show that the coalition having access to the set $B_h \cup \{B_{0h(\psi+1)}\}$ is equivalent to C knowing all of the signing information for U_h, namely the set

$$\{D_{0k\ell h} : 0 \leq k \leq \psi,\ 0 \leq \ell \leq \gamma\}.$$

First note that these two sets are both of cardinality $(\psi + 1)(\gamma + 1)$.

It is easy to see that the equations in $B_h \cup \{B_{0h(\psi+1)}\}$ may be written as a linear combination of the equations in $\{D_{0k\ell h} : 0 \leq k \leq \psi, 0 \leq \ell \leq \gamma\}$. To see that $B_h \cup \{B_{0h(\psi+1)}\}$ suffices to derive the signing information of U_h, note that there are $\psi + 1$ equations $\{B_{0hk'} : 1 \leq k' \leq \psi + 1\}$ in the $\psi + 1$ unknowns $\{D_{0k0h} : 0 \leq k \leq \psi\}$. (The linear independence of these equations is guaranteed so long as the messages chosen for the sample signatures from U_h are distinct.)

Now suppose C has signing key exposure for U_h instead of master key exposure. Let

$$B'_h = \{E_{0kt_{h_d}} : 0 \leq k \leq \psi,\ 1 \leq d \leq \gamma\} \cup \{B_{0hk'} : 1 \leq k' \leq \psi\}.$$

It is easy to see that the equations in $B_h \cup \{B_{0h(\psi+1)}\}$ may be written as a linear combination of the equations in $\{D_{0k\ell h} : 0 \le k \le \psi, 0 \le \ell \le \gamma\}$ and that these two sets have the same cardinality. To see that $B_h \cup \{B_{0h(\psi+1)}\}$ suffices to derive the signing information of U_h, note that there are $\psi + 1$ equations $\{B_{0hk'} : 1 \le k' \le \psi + 1\}$ and $\gamma(\psi + 1)$ equations $\{E_{0kt_{h_d}} : 0 \le k \le \psi, 1 \le d \le \gamma\}$ in the $(\psi + 1)(\gamma + 1)$ unknowns $\{D_{0k\ell h} : 0 \le k \le \psi, 0 \le \ell \le \gamma\}$. (The linear independence of these equations is guaranteed so long as the messages chosen for the sample signatures from U_h are distinct.) □

The following lemma completes the result:

Lemma A.7. *The coefficient matrix formed from the equations*

$$\{C_{ik\ell h} : 0 \le i \le n-1,\ 0 \le k \le \psi,\ 0 \le \ell \le \gamma,\ 1 \le h \le \omega\} \cup \{D_{0k\ell h} : 0 \le k \le \psi,\ 0 \le \ell \le \gamma,\ 1 \le h \le n\}$$

has nonzero determinant.

Proof. The coefficient matrix E is a block matrix of the form

$$E = \left[\begin{array}{c:c} A & 0 \\ \hdashline C & D \end{array}\right],$$

where A and D are square matrices. Thus the determinant of the coefficient matrix, $\det(E)$, is defined by $\det(E) = \det(A)\det(D)$. We show that $\det(E) \neq 0$.

Here the submatrix

$$[\, A \,\vdots\, 0 \,]$$

is derived from the equations $\{D_{0k\ell h} : 0 \le k \le \psi, 0 \le \ell \le \gamma, 1 \le h \le n\}$, where

$$A = \begin{bmatrix} V_n & 0 & \cdots & 0 \\ 0 & V_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & V_n \end{bmatrix}$$

is a diagonal matrix with $(\psi + 1)(\gamma + 1)$ Vandermonde matrices V_n on the diagonal. That is, we have

$$V_n = \begin{bmatrix} 1 & U_1 & \cdots & U_1^{n-1} \\ 1 & U_2 & \cdots & U_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & U_n & \cdots & U_n^{n-1} \end{bmatrix}.$$

To see that A is invertible, note that $\det(A) = \det(V_n)^{(\psi+1)(\gamma+1)} \neq 0$.

The submatrix

$$[\, C \,\vdots\, D \,]$$

is derived from the equations $\{C_{ik\ell h} : 0 \le i \le n-1, 0 \le k \le \psi, 0 \le \ell \le \gamma, 1 \le h \le \omega\}$. The matrix D is defined by

$$D = \begin{bmatrix} v_{1,1}\mathrm{I} & v_{1,2}\mathrm{I} & \cdots & v_{1,\omega}\mathrm{I} \\ v_{2,1}\mathrm{I} & v_{2,2}\mathrm{I} & \cdots & v_{2,\omega}\mathrm{I} \\ \vdots & \vdots & \ddots & \vdots \\ v_{\omega,1}\mathrm{I} & v_{\omega,2}\mathrm{I} & \cdots & v_{\omega,\omega}\mathrm{I} \end{bmatrix},$$

where I is the $n(\psi + 1)(\gamma + 1) \times n(\psi + 1)(\gamma + 1)$ identity matrix.

The fact that $\det(D) \neq 0$ follows immediately from the linear independence of the coalition's verification keys $\{\vec{v_h} : 1 \le h \le \omega\}$. □

Acknowledgment: The authors would like to acknowledge the feedback of the referees.

Funding: Douglas R. Stinson is supported by NSERC grant 203114-11.

References

[1] E. F. Brickell and D. R. Stinson, Authentication codes with multiple arbiters (extended abstract), in: *Advances in Cryptology* (EUROCRYPT '88), Lecture Notes in Comput. Sci. 330, Springer, Berlin (1988), 51–55.

[2] D. Chaum and S. Roijakkers, Unconditionally secure digital signatures, in: *Advances in Cryptology* (CRYPTO '90), Lecture Notes in Comput. Sci. 537, Springer, Berlin (1991), 206–214.

[3] Y. Desmedt, Y. Frankel and M. Yung, Multi-receiver/multi-sender network security: Efficient authenticated multicast/feedback, in: *IEEE International Conference on Computer Communications* (INFOCOM '92), IEEE Press, Piscataway (1992), 2045–2054.

[4] Y. Desmedt and M. Yung, Abritrated unconditionally secure authentication can be unconditionally protected against arbiter's attacks (extended abstract), in: *Advances in Cryptology* (CRYPTO '90), Lecture Notes in Comput. Sci. 537, Springer, Berlin (1991), 177–188.

[5] Y. Dodis, J. Katz, S. Xu and M. Yung, Key-insulated public key cryptosystems, in: *Advances in Cryptology* (EUROCRYPT 2002), Lecture Notes in Comput. Sci. 2332, Springer, Berlin (2002), 65–82.

[6] Y. Dodis, J. Katz, S. Xu and M. Yung, Strong key-insulated signature schemes, in: *Public Key Cryptography* (PKC 2003), Lecture Notes in Comput. Sci. 2567, Springer, Berlin (2003), 130–144.

[7] G. Hanaoka, J. Shikata, Y. Zheng and H. Imai, Unconditionally secure digital signature schemes admitting transferability, in: *Advances in Cryptology* (ASIACRYPT 2000), Lecture Notes in Comput. Sci. 1976, Springer, Berlin (2000), 130–142.

[8] G. Hanaoka, J. Shikata, Y. Zheng and H. Imai, Efficient and unconditionally secure digital signatures and a security analysis of a multireceiver authentication code, in: *Public Key Cryptography* (PKC 2002), Lecture Notes in Comput. Sci. 2274, Springer, Berlin (2002), 64–79.

[9] Y. Hara, T. Seito, J. Shikata and T. Matsumoto, Unconditionally secure blind signatures, in: *Information Theoretic Security* (ICITS 2007), Lecture Notes in Comput. Sci. 4883, Springer, Berlin (2009), 23–43.

[10] T. Johansson, On the construction of perfect authentication codes that permit arbitration, in: *Advances in Cryptology* (Crypto '93), Lecture Notes in Comput. Sci. 773, Springer, Berlin (1994), 343–354.

[11] T. Johansson, Further results on asymmetric authentication schemes, *Inform. and Comput.* **151** (1999), no. 1–2, 100–133.

[12] R. Safavi-Naini, L. McAven and M. Yung, General group authentication codes and their relation to "unconditionally-secure signatures", in: *Public Key Cryptography* (PKC 2004), Lecture Notes in Comput. Sci. 2947, Springer, Berlin (2004), 231–247.

[13] R. Safavi-Naini and H. Wang, Broadcast authentication in group communication, in: *Advances in Cryptology* (ASIACRYPT '99), Lecture Notes in Comput. Sci. 1716, Springer, Berlin (1999), 399–411.

[14] T. Seito, T. Aikawa, J. Shikata and T. Matsumoto, Information-theoretically secure key-insulated multireceiver authentication codes, in: *Progress in Cryptology* (AFRICACRYPT 2010), Lecture Notes in Comput. Sci. 6055, Springer, Berlin (2010), 148–165.

[15] T. Seito and J. Shikata, Information-theoretically secure key-insulated key-agreement, in: *IEEE Information Theory Workshop* (ITW 2011), IEEE Press, Piscataway (2011), 287–291.

[16] J. Shikata, G. Hanaoka, Y. Zheng and H. Imai, Security notions for unconditionally secure signature schemes, in: *Advances in Cryptology* (EUROCRYPT 2002), Lecture Notes in Comput. Sci. 2332, Springer, Berlin (2002), 434–449.

[17] G. J. Simmons, Message authentication with arbitration of transmitter/receiver disputes, in: *Advances in Cryptology* (EUROCRYPT '87), Lecture Notes in Comput. Sci. 304, Springer, Berlin (1987), 151–165.

[18] G. J. Simmons, A cartesian product construction for unconditionally secure authentication codes that permit arbitration, *J. Cryptology* **2** (1990), 77–104.

[19] C. M. Swanson and D. R. Stinson, Unconditionally secure signature schemes revisited, in: *Information Theoretic Security* (ICITS 2011), Lecture Notes in Comput. Sci. 6673, Springer, Berlin (2011), 100–116.