Strong Anonymous Signatures

Rui Zhang and Hideki Imai

Research Center for Information Security (RCIS),
National Institute of Advanced Industrial Science and Technology (AIST), Japan
{r-zhang,h-imai}@aist.go.jp

Abstract. The notion of anonymous signatures has recently been formalized by [18], which captures an interesting property that a digital signature can sometimes hide the identity of the signer, if the message is hidden from the verifier. However, in many practical applications, e.g., an anonymous paper review system mentioned in [18], the message for anonymous authentication is actually known to the verifier. This implies that the effectiveness of previous anonymous signatures may be unjustified in these applications. In this paper, we extend the previous models, and develop a related primitive called strong anonymous signatures. For strong anonymous signatures, the identity of the signer remains secret even if the challenge message is chosen by an adversary. We then demonstrate some efficient constructions and prove their security in our model.

keywords: digital signatures, anonymity

1 Introduction

The notion of anonymous signatures was proposed by [18] in PKC'06, which captures the property that a digital signature can sometimes hide the identity of the signer, provided that the message is hidden from the verifier. Many applications of anonymous signature were mentioned, especially to achieve simple anonymous authentication with ordinary digital signatures, like key exchange protocols or auction systems.

Let's take a closer look at the primitive by an example, an anonymous review system mentioned in [18]. In the submission phase, a paper (the message) is submitted together with an anonymous signature. Then in the review phase, because of the anonymity of the signature, the identities of the authors are not decidable. Finally, if his paper is accepted, the author (signer) reveals the identity by disclosing the message/signature pair, authenticating that he submitted the paper. If the paper is rejected, he keeps silent, which will leave the submission remaining anonymous.

A straightforward adoption of the anonymous signature in such a system may be problematic, because the message and the signature are both public, it is uniquely verifiable the identity of a particular signer. We remark that the same problem happens in many similar applications. The authors of [18] proposed a countermeasure in their paper, but no formal analysis could be found. Furthermore, it is not evident whether previous anonymous signatures provides anonymity if (part of) the message is known.

To motivate this work a little more, consider an another scheme given in [11]. Denote m as a message, σ as the corresponding signature signature (unnecessarily anonymous). Denote $\mathsf{Ext}(\cdot)$ as an extractor. Informally speaking, an extractor takes a high entropy source as input and produces an almost uniformly random sequence. It was claimed that $\mathsf{Ext}(m)\oplus\sigma$ is an anonymous signature according to the model [18, 11]. It is easy to see that if m is revealed, σ is uniquely verifiable. On the other hand, by using the leftoverhash lemma, in order to have 128-bit security, the message m should have at least about 400-bit min-entropy, which is not always satisfied in some applications. We stress that the above construction [11] is not claimed to be secure when m is of low entropy to the verifier.

To summarize, the security of the previous constructions may sometimes be problematic for messages with low entropy to the adversary, however, the assumption on message entropy is not justified in practical applications for anonymous signatures. It is natural to ask the following question: *Can we have a suitably strong model of anonymous signatures (where messages can be of any distribution) and efficient constructions secure in this model*? In this paper, we try to give an answer.

1.1 Related Work

The anonymity of public key encryption was formulated in [2]. As an analogy of anonymity of encryption schemes, the notion of anonymity of digital signatures was formulated by Yang et al. [18]. The first constructions of [18] were based on specific signature schemes and the security of the constructions have to resort to random oracles. Subsequently, Fischlin proposed some constructions of anonymous signatures without random oracles [11]. We note that we found similar constructions independently, and have presented in local meetings before [11] was published.

Many different primitives have been proposed for the anonymity of signatures [6, 8, 9, 7, 13, 14, 16]. Most of these known solutions are equipped with powerful anonymity or traceability, while as a trade-off they usually require complicated setups or inefficient computations. For some specific tasks needing only “handy” anonymity, e.g., the anonymous paper review system, these may be too expensive.

1.2 Our Contributions

We view our main contributions as pointing out the limitations of the previous models and giving simple solutions. We first point out the limitations of the previous security definitions for some specific applications, and propose an extended security model. In most cases, the new security definition is stronger: all the schemes secure in the new model are still secure in the previous one, while the converse may not hold. In some cases, when the message is hidden from a distinguisher, and the new security model degenerates to the previous one, so that the previous constructions can be adopted.

Next we demonstrate several efficient generic constructions that satisfy our new model without changing the internal structure of underlying primitives. Our first construction, inspired by [18], applies an efficient pre-processing to the message before signing. However, this suffices to turn a (weak) anonymous signature into a strong one. We also give some other constructions using pseudorandom generators, and ordinary (unable) digital signatures. Somehow surprisingly, our result actually show that the strong anonymity can be achieved with the same set of additional assumptions as previous constructions.

All the constructions are simple in concept, generic and efficient. As a result, a strong anonymous signature is easy to achieve and efficient based on ordinary digital signatures.

We also discuss possible extensions of our constructions. An immediate application of our result is a rigorous proof of a previous conjecture mentioned in [18].

As an independent interest, we highlight a new notion called collision resistant exposure resilient functions (CR-ERFs). An exposure resilient function (ERF) [5] was a primitive whose output looks random even if most of its input is known by an adversary. An ERF is a useful tool to protect the leakage of secret keys. However, the collision resistance of inputs were not considered in the original formulation of ERFs. Certainly, this suffices for some applications, however, we argue it may be not sufficient for specific applications: Two different inputs may produce the same output sequence, though no efficient adversary cannot distinguish the output sequence from uniformly random ones. In this paper, we give a formal dentition of ERFs with collision resistance. We further give a simple construction assuming random oracles.

1.3 Organization of the Paper

In Section 2, we review some useful notations and notions. In Section 3, we show how to convert any weak anonymous signature schemes into a strong one. We go on to show constructions of strong anonymous signature from more fundamental primitives: namely unforgeable digital signatures and pseudorandom generators in Section 4. Finally, we give the conclusion in Section 6.

2 Preliminary

In this section, we give some notations and definitions.

Notations. If S is a set then $s \leftarrow S$ denotes the operation of picking an element s of S uniformly at random. We write $z \leftarrow \mathcal{A}(x, y, \ldots)$ to indicate that $\mathcal{A}$ is an algorithm with inputs $(x, y, \ldots)$ and an output z. Denote $x \| y$ as the string concatenation of x and y. If $k \in \mathbb{N}$, a function $f(k)$ is negligible if $\exists\ k_0 \in \mathbb{N}$, $\forall\ k > k_0$, $f(k) < 1/k^c$, where $c > 0$ is a constant.

2.1 Digital Signature

A signature scheme Σ consists of three algorithms $\Sigma = (\mathsf{G}, \mathsf{S}, \mathsf{V})$. The randomized key generation algorithm G takes a security parameter k, and generates signing key sk and verification key vk, denoted as $(pk, sk) \leftarrow \mathsf{G}(k)$. The possibly randomized signing algorithm S takes as inputs sk, a message $m \in \mathcal{M}$ where $\mathcal{M}$ is the message space, and an auxiliary input s (of a certain distribution), outputs a signature σ, denoted as $\sigma \leftarrow \mathsf{S}(sk, m, s)$. The deterministic verification algorithm V takes as inputs vk, m, s and σ, and outputs a symbol $\beta \in (0, 1)$, denoted as $\beta \leftarrow \mathsf{V}(vk, m, \sigma)$. We require that $\forall\ (sk, vk) \leftarrow \mathsf{G}(k)$, $\forall\ m \in \mathcal{M}$, $1 = \mathsf{V}(vk, m, s, \mathsf{S}(sk, m, s))$. If s is an constant string, for simplicity, we can be omitted from the input of the algorithms, because it can be viewed a part of the verification key (and the signing key) anyway.

In the above formulation, we have introduced an auxiliary input s into the signing and verification algorithms. We note that the auxiliary input is harmless to the model. If s is an empty string, the syntax degenerates to the classical model of digital signatures. On the other hand, the existence of this auxiliary input enables us to define a stronger version of anonymity: all except s can be revealed to an adversary, while the identity of the signer is still unknown.

Anonymity. A first flavor of anonymity, which we call weak anonymity (wa) [18, 11], states that a signature should not leak the identity of its signer, if the message (chosen by the challenger) is hidden from the verifier, even under adaptive chosen message attack (CMA).

Definition 1 **(**WA-CMA). *Denote $\mathcal{M}$ as the message space and $\mathcal{SO}$ as a signing oracle that returns the corresponding signature σ on a signing query m. Denote ξ is an empty string. We say a signature scheme is weakly anonymous against chosen message attack (*WA-CMA*) if any probabilistic polynomial time (PPT) adversary $\mathcal{A}$'s advantage is negligible in the following experiment.*

$$\begin{aligned}\mathrm{Adv}^{\mathsf{wa\text{-}cma}}_{\Sigma,\mathcal{A}} = |\Pr[(vk_0, sk_0) \leftarrow \mathsf{G}(k); (vk_1, sk_1) \leftarrow \mathsf{G}(k); m \leftarrow \mathcal{M}; b \leftarrow \{0, 1\};\\ \sigma \leftarrow \mathsf{S}(sk_b, m, \xi); b' \leftarrow \mathcal{A}^{\mathcal{SO}}(vk_0, vk_1, \sigma) : b' = b] - 1/2|\end{aligned}$$

In the above definition, we insist that the auxiliary input for the signing algorithm is an empty string to be compatible with the previous definitions. However, as pointed at the beginning, weak anonymity suffices in some cases, but it may not be suitable for all applications of anonymous signatures.

We want to give a "properly" strong definition for anonymous signatures, by introducing the following modifications to the previous model. The first modification is that the message, together with the signature, is also presented to a distinguisher. Moreover, in the previous model, the message is chosen by the challenger, however, it may seem a little strange, since the adversary's power is limited to be passive, while it can access the signing oracle adaptively. We then allow the challenge message can be chosen adaptively by the adversary. We call our new definition strong anonymity against chosen message attack (SA-CMA).

Definition 2 **(**SA-CMA). *Denote $\mathcal{M}$ as the message space and $\mathcal{SO}$ as a signing oracle that returns the corresponding signature σ on a signing query m. Denote s as a random number with certain probability distribution. Let st be the state information for $\mathcal{A}$. We say a signature scheme is anonymous if any PPT adversary $\mathcal{A}$'s advantage is negligible in the following experiment.*

$$\begin{aligned}\mathrm{Adv}^{\mathsf{sa}}_{\Sigma,\mathcal{A}}(k) = |\Pr[(vk_0, sk_0) \leftarrow \mathsf{G}(k); (vk_1, sk_1) \leftarrow \mathsf{G}(k); (m, st) \leftarrow \mathcal{A}^{\mathcal{SO}}(vk_0, vk_1);\\ b \leftarrow \{0, 1\}; \sigma \leftarrow \mathsf{S}(sk_b, m, s); b' \leftarrow \mathcal{A}^{\mathcal{SO}}(\sigma, st) : b' = b] - 1/2|\end{aligned}$$

Unforgeability. Here we consider two flavors of unforgeability, i.e., weak unforgeability (UF) [12] and strong unforgeability (sUF) [1]. Let $\Sigma = (\mathsf{G}, \mathsf{S}, \mathsf{V})$ be a signature scheme. Let $\mathcal{A}$ be an adversary and k be a security parameter, respectively. Denote $\mathcal{L}$ as the transcript containing all the interactions between $\mathcal{A}$ and $\mathcal{SO}$, where $\mathcal{SO}$ is a signing oracle that for an input message m, returns a corresponding signature σ. Consider the success probability of $\mathcal{A}$ in the following two cases.

Definition 3 (UF-CMA). *We say Σ is (t, ϵ)-*UF-CMA *secure if for any $\mathcal{A}$ in time bound t, $\mathcal{A}$'s success probability is at most ϵ in the following experiment. Especially, we say that Σ is* UF-CMA *secure if ϵ is negligible.*

$$\mathrm{Suc}^{\text{uf-cma}}_{\Sigma,\mathcal{A}}(k) = \Pr[(vk, sk) \leftarrow \mathsf{G}(k); (m^*, s^*, \sigma^*) \leftarrow \mathcal{A}^{\mathcal{SO}}(vk) : \\ \mathsf{V}(vk, m^*, s^*, \sigma^*) = 1 \wedge m^* \notin \mathcal{L}]$$

Definition 4 (sUF-CMA). *We say Σ is (t, ϵ)-*sUF-CMA *secure if for any $\mathcal{A}$ in time bound t, $\mathcal{A}$'s success probability is at most ϵ in the following experiment. Especially, we say that Σ is* sUF-CMA *secure if ϵ is negligible.*

$$\mathrm{Suc}^{\text{suf-cma}}_{\Sigma,\mathcal{A}}(k) = \Pr[(vk, sk) \leftarrow \mathsf{G}(k); (m^*, s^*, \sigma^*) \leftarrow \mathcal{A}^{\mathcal{SO}}(vk) : \\ \mathsf{V}(vk, m^*, s^*, \sigma^*) = 1 \wedge (m^*, s^*, \sigma^*) \notin \mathcal{L}]$$

We believe the introduction of the auxiliary input s is only conceptual and harmless to the model. If we modify the syntax, i.e., viewing s as a part of the signature, then the new definitions degrade to the conventional unforgeability [12, 1].

Finally, if a signature scheme is both unforgeable and anonymous, we say it is a secure anonymous signature scheme.

2.2 Collision Resistant-Exposure Resilient Function (CR-ERF)

The notion of exposure resilient function (ERF) was proposed in [5] by Canetti et al., which deals with gradual leakage of secret keys. An ERF is a deterministic function whose output appears random even if all of the bits of input are known. The security definition of ERF has several flavors: perfect, statistical and computational settings. In the computational setting, it is known that secure computational ERFs exist if and only if oneway functions exist [5].

We need a slightly stronger primitive for our construction, called collision resistant exposure resilient function (CR-ERF). The difference of CR-ERFs from ERFs is that it is also infeasible to find a collision for the same output. It is sufficient to require the output of a CR-ERF to be pseudorandom, since our main application is strong anonymous signatures.

Definition 5 (CR-ERF). *Denote* $\mathsf{U}(n)$ *as an operation of sampling n-bits from uniform distribution. Denote st as state information. Denote a deterministic polynomial time computable function* $f(x) : \{0,1\}^{n_0} \rightarrow \{0,1\}^{n_1}$. *Consider adversary* $\mathcal{A}$*'s advantage of the following two experiments:*

$$\mathrm{Adv}^{\mathsf{ind}}_{\mathsf{CR\text{-}ERF},\mathcal{A}}(k) = |\Pr[r \leftarrow \{0,1\}^{\ell}; (x, st) \leftarrow \mathcal{A}(k); X_0 \leftarrow f(x\|r); \\ X_1 \leftarrow \mathsf{U}(n_1); b \leftarrow \{0,1\}; b' \leftarrow \mathcal{A}(X_b, st) : b = b'] - 1/2|$$

$$\mathrm{Adv}^{\mathsf{cr}}_{\mathsf{CR\text{-}ERF},\mathcal{A}}(k) = \Pr[(x_0, r_0, x_1, r_1) \leftarrow \mathcal{A}(k) : f(x_0\|r_0) = f(x_1\|r_1)]$$

We say a CR-ERF is secure, if any probabilistic polynomial time (PPT) adversary's advantage in each of the above games is negligible in k.

Note that the above definition on indistinguishability is weak, since the positions of bits are fixed in the input for the ERF. However, we can see later this is sufficient for our purpose. We give a simple the construction of a CR-ERF in Section 3.2.

2.3 Pseudorandom Generator (PRG)

A cryptographically secure pseudorandom generator is a deterministic function $G : \{0,1\}^{n_0} \rightarrow \{0,1\}^{n_1}$ that satisfies two properties:

1. Expansion: Namely, $n_1 > n_0$.
2. Pseudorandomness: Namely, any PPT algorithm $\mathcal{A}$'s advantage is negligible in the following experiment:

$$\mathrm{Adv}^{\mathsf{ind}}_{\mathcal{A}}(k) = \Pr[r \leftarrow \{0,1\}^{n_0}; x_1 \leftarrow G(r); x_2 \leftarrow \mathsf{U}(n_1); \\ b \leftarrow \{0,1\}; b' \leftarrow \mathcal{A}(x_b) : b = b'] - 1/2|$$

PRG is an important primitive, which is well-studied with many practical constructions.

3 From Weak Anonymity to Strong Anonymity

In this section, we present a generic construction of a strong anonymous signature from any weakly anonymous (WA) signature using collision resistant exposure resilient functions (CR-ERFs). The construction is of a black-box manner, such that all previous constructions of weak anonymous signatures can be reused. The core idea is to improve WA-CMA secure signatures with a CR-ERF function. Actually as we have demonstrated in the beginning, the adversary is able to see the message, thus there is no enough entropy for the previous constructions. On the other hand, observe that an CR-ERF outputs pseudorandom sequences, even when part of the its input is leaked to the adversary. We consider to utilize the classical "hash-then-sign" paradigm, namely, pre-process the message with a CR-ERF, then sign on the output of the CR-ERF. Actually, this simple idea suffices.

3.1 The Construction

Instead of signing directly on a message m, first apply a CR-ERF f to m with a random number s, then sign on $f(m\|s)$ to get the signature σ. We claim the signature scheme acquired with (σ, m) being public and s kept secret is a secure strong anonymous signature. We elaborate the construction below.

Suppose $\Sigma' = (\mathsf{G}', \mathsf{S}', \mathsf{V}')$ is an anonymous signature (associated with message space $\mathcal{M}$) in the sense of [18, 11].

- **Key Generation** $\mathsf{G}(k)$: The algorithm calls $\mathsf{G}'(k)$, where k is a security parameter, and returns vk, the verification key, and sk, the secret signing key. Suppose $f : \mathcal{M} \times \{0,1\}^{\ell} \rightarrow \{0,1\}^{len}$ is a CR-ERF, where len is a parameter defined by the public key of vk. The public verification key for Σ is (vk, f), and secret key is (sk, f).
- **Signing** $\mathsf{S}(sk, m, s)$: The algorithm returns σ, where $\sigma = \mathsf{S}'(sk, f(m\|s))$, where sk is a secret signing key, $m \in \mathcal{M}$ is a message and s is a random number of length ℓ.
- **Verification** $\mathsf{V}(vk, m, s, \sigma)$: The algorithm returns a bit β, where $\beta = \mathsf{V}'(vk, f(m\|s), \sigma)$.

Theorem 1. *The above construction is a strong anonymous signature, assuming $f(m\|s)$ is a CR-ERF, with m be the public part and s being the secret part.*

Proof. The correctness of the scheme is obviously seen. We next claim the above construction achieves exactly the same level of unforgeability as the underlying weak anonymous signature scheme. To see this, notice that our construction in fact follows the classical hash-then-sign paradigm, and recall that the CR-ERF is collision resistant, then any forger needs to either break the collision resistance or the unforgeability of the underlying signature to succeed.

We focus on the strong anonymity. It will be enough to show any adversary against the above signature scheme can be converted into either an adversary against the CR-ERF, or an adversary against Σ'. We demonstrate the proof in the game hopping style [15]. We will design a sequence of games. Denote E_i as the event that the adversary succeed in a specific game Game i. We will bound the probability difference of each game, and finally reach our result. First let us review a useful lemma here.

Lemma 1 ([15]). *Let A, B, F be events defined in some probability distribution, and suppose that $A \wedge \neg F \Leftrightarrow B \wedge \neg F$. Then*

$$|\Pr[A] - \Pr[B]| \le \Pr[F].$$

Game 0 : The same as Definition 2. We have $\mathrm{Adv}^{\mathsf{sa}}_{\Sigma,\mathcal{A}}(k) = \Pr[E_0]$.

Game 1 : Instead of computing the CR-ERF as defined, the challenger, for the challenge, instead of computing $f(m\|s)$, it picks a random number $e \leftarrow \{0,1\}^{len}$. We claim

$$\Pr[E_1] - \Pr[E_0] \le \mathrm{Adv}^{\mathsf{erf}}_{\Sigma,\mathcal{A}}(k) \tag{1}$$

To see this, one can build an adversary that distinguishes the output of a CR-ERF from uniformly random sequence by success probability difference between Game 0 and Game 1, which falls into a standard hybrid argument.

We then claim

$$\Pr[E_1] \leq \mathrm{Adv}^{\mathsf{wa}}_{\Sigma,\mathcal{A}}(k) \tag{2}$$

Notice that the challenge is of the form (m^*, σ^*), while $e \leftarrow \{0,1\}^{len}$ and $\sigma^* = \mathsf{S}'(sk_b, e^*)$ is computed using the underlaying signature scheme. Due to the anonymity of the underlying signature scheme Σ', the distribution of (m^*, σ^*) is indistinguishable from (m^*, σ'), where $\sigma' \leftarrow \mathsf{S}'(sk_{1-b}, e^*)$. One can conclude that any adversary of Game 1 against Σ can be converted to an adversary to Σ'. Thus $\mathcal{A}$'s advantage is at most $\mathrm{Adv}^{\mathsf{wa}}_{\mathcal{A}}(k)$. Summarizing Eqs. (1), (2), we conclude

$$\mathrm{Adv}^{\mathsf{sa}}_{\mathcal{A}}(k) \leq \mathrm{Adv}^{\mathsf{erf}}_{\mathcal{A}}(k) + \mathrm{Adv}^{\mathsf{wa}}_{\mathcal{A}}(k)$$

This completes the proof of Theorem 1. □

3.2 A Simple CR-ERF Assuming Random Oracles

Here we present a construction of CR-ERF assuming random oracles. Let r be a random number, whose length is ℓ-bits. Let $H : \{0,1\}^* \times \{0,1\}^{\ell} \rightarrow \{0,1\}^n$ be a cryptographically secure hash function, which will be modeled as a random oracle in the following analysis. For simplicity, we assume $\ell \geq 2k$ and $n \geq 2k$, where k is a security parameter.

Lemma 2. *$H(m||s)$ is a CR-ERF, with s kept secret from the adversary.*

Proof. The proof is very simple. Denote AskH the event that $m||s$ has already been queried to H by an adversary. Within Q_H random oracle queries, a collision of the output by the random oracle is at most $\Pr[\mathsf{AskH}] \leq Q_H \cdot 2^{-\ell}$. On the other hand, if $m||s$ is not queried to H, the probability of distinguishing the output of H from another uniform random sequence is exactly 1/2. Thus we have:

$$\begin{aligned}\mathrm{Adv}^{\mathsf{cr}}_{\mathsf{CR\text{-}ERF},\mathcal{A}}(k) &= 1/2\Pr[\overline{\mathsf{AskH}}] + \Pr[\mathsf{AskH}] - 1/2\\ &= 1/2(1 - Q_H 2^{-\ell}) + Q_H 2^{-\ell} - 1/2\\ &= Q_H 2^{-(\ell+1)}\end{aligned}$$

which is negligible for sufficiently large ℓ.

On the other hand, for collision resistance, if H is a random oracle, the probability of get collisions via Q_H queries is at most $Q_H \cdot 2^{-\ell}$. Combine all these discussions, we conclude that our construction is a secure CR-ERF.

4 Constructions from Unforgeable Signatures

In this section, we present two constructions of strong anonymity signatures from ordinary signature schemes. Let us first recall the idea of [11] here. An anonymous signature scheme mask an ordinary signature scheme with some randomness, and later removes this randomness for verification. Particularly, [11] has done this by extracting randomness from the message source deterministically.

On the other hand, one has to assume the source has enough min-entropy, so that an extractor can be used to extract the necessary amount randomness from the source. Since the extractor is deterministic given the message, thus the signature is uniquely verifiable when the message is shown later. However, as we have argued in the beginning, if the message contains little entropy, certainly, the constructions of [11] are not anonymous.

4.1 A Construction with Weak Unforgeability

Actually, building a strong anonymous signature with weak unforgeability is very easy. Here we give a simple construction. We remark that this idea is not taken from [11], since we hade presented it in local meetings long before [11] was published. Denote $\Sigma' = (\mathsf{G}', \mathsf{K}', \mathsf{V}')$ is a $\mathsf{UF\text{-}CMA}$ secure signature scheme. Let $\Sigma = (\mathsf{G}, \mathsf{K}, \mathsf{V})$ as follows:

Key Generation $\mathsf{G}(k)$**:** The algorithm calls $\mathsf{G}'(k)$, where k is a security parameter, and returns (vk, h), the verification key, and sk, the secret signing key. Here $h : \{0,1\}^\ell \to \{0,1\}^{len}$ is a pseudorandom generator, where ℓ is the length of the seed, and len is a constant indicating the bit length of a signature. The public verification for Σ is (vk, h), and the secret signing key is (sk, h).

Signing $\mathsf{S}(sk, m, s)$**:** The algorithm returns $(\sigma \oplus h(s), s)$, where $\sigma = \mathsf{S}'(sk, m)$, where sk is a secret signing key, m is a message and s is a random number of length ℓ.

Verification $\mathsf{V}(vk, m, s, e)$**:** The algorithm returns a bit β, where $\beta = \mathsf{V}'(vk, m, e \oplus h(s))$.

Actually, our construction offers a possibly better performance than [11], which is only weakly anonymous.

The intuition of this construction is to use external randomness, namely, let the signature be $\sigma \oplus h(s)$, where $\sigma \leftarrow \mathsf{S}'(sk, m)$, $h(\cdot)$ is a pseudorandom generator and s is a random seed. For verification, just reveal m and s. We emphasize this simple construction actually achieves strong anonymity and (weak) unforgeability. The intuition is that without knowing s, $\sigma \oplus h(s)$ is actually pseudorandom, thus the strong anonymity is achieved. On the other hand, given a valid forged signature on a message unsigned previously, one can extract the a forgery for the underlying signature scheme.

Theorem 2. *The above construction is a secure strong anonymous signature with weak unforgeability.*

Proof. For weak unforgeability ($\mathsf{UF\text{-}CMA}$), it suffices to show any forger $\mathcal{A}$ for the above signature scheme Σ can be transformed into a forger $\mathcal{B}$ to the underlying signature scheme Σ'. First, for setup, $\mathcal{B}$ gives its own target verification key and a pseudorandom generator

h to $\mathcal{A}$. It is easy to see this is a correct public key for Σ. Then we claim that all the signing queries can be correctly answered. To see this, a simulator just relays the message m to its own signing oracle. Once it gets the signature σ' from its oracle, it chooses s at random and set $\sigma = \sigma \oplus h(s)$. It is verifiable that this is a correct signature for m according to the definition of Σ. Next, when a forger terminates and outputs its forgery for Σ, one can extract from any forgery (m^*, s^*, σ^*) a valid signature for Σ', namely $(m^*, \sigma^* \oplus h(s^*))$, where m^* is a message never queried to the signing oracle of the underlying signature scheme Σ'. It is also verifiable that the correctness of this forgery for Σ'. We conclude that $\mathcal{B}$'s success probability is exactly that of $\mathcal{A}$.

For anonymity, we show how to transform any distinguisher $\mathcal{A}$ for Σ to a distinguisher of a distinguisher $\mathcal{B}$ for the pseudorandom generator h. For setup, $\mathcal{B}$ runs the key generation algorithm, generating two pairs of verification/signing keys (vk_0, sk_0) and (vk_0, sk_0) using G'. $\mathcal{B}$ then gives (vk_0, h) and (vk_1, h) to $\mathcal{A}$ as the public key. To remark, $\mathcal{B}$ can handle any signing query easily, since it has the signing key. When $\mathcal{A}$ chooses a message m^* and hands it to $\mathcal{B}$, and $\mathcal{B}$ will select $b \leftarrow \{0, 1\}$. It then sets $\sigma' \oplus T$, where $\sigma' = \mathsf{S}'(sk_b, m^*)$ and T of length *len* is its own challenge. When $\mathcal{A}$ outputs its guess b', $\mathcal{B}$ outputs 1 (pseudorandom) if $b = b'$ and 0 (truly random) if $b \neq b'$ as its answer.

If T is truly random, the information of b perfectly hiding, and $\mathcal{A}$ can only win the game with probability $1/2$. We conclude $\mathcal{A}$ can gain advantage (than random guess) in the game only if T is not truly random. Thus the advantage of $\mathcal{B}$ is exactly that of $\mathcal{A}$. □

Finally, we remark that the generic construction admits *tight* security reductions to the primitives.

4.2 A Construction with Strong Unforgeability

We slightly modify the generic construction in 4.1 to achieve strong unforgeability. The main difference is that we require the underlying signature scheme to have strong unforgeability. Actually this is easy if the random seed s is signed together by the signing algorithm. The intuition is that if the underlying signature scheme is strongly unforgeable, the integrity of each message/signature pair is maintained together with every s. Denote $\Sigma' = (\mathsf{G}', \mathsf{K}', \mathsf{V}')$ is a sUF-CMA secure signature scheme.

Key Generation $\mathsf{G}(k)$**:** The algorithm calls $\mathsf{G}'(k)$, where k is a security parameter, and returns vk, the verification key, and sk, the secret signing key. Here $h : \{0, 1\}^\ell \rightarrow \{0, 1\}^{len}$ is a pseudorandom generator, where *len* is a constant indicating the bit length of a signature. The public verification for Σ is (vk, h), and the secret signing key is (sk, h).

Signing $\mathsf{S}(sk, m)$**:** The algorithm returns $(\sigma \oplus h(s), s)$, where $\sigma = \mathsf{S}'(sk, m\|s)$, where sk is a secret signing key, m is a message and s is a random number of length ℓ.

Verification $\mathsf{V}(vk, m, s, e)$**:** The algorithm returns a bit β, where $\beta = \mathsf{V}'(vk, m, e \oplus h(s))$.

Theorem 3. *The above construction is a secure strong anonymous signature with strong unforgeability.*

Proof. The proof mostly repeats that of Theorem 2 except following some subtle points. The anonymity is achieved since the signature is masked by a pseudorandom sequence (s is not revealed in the anonymity game).

For strong unforgeability, as usual, a simulator $\mathcal{B}$ against Σ' picks a pseudorandom generator h and its own challenge signature verification key vk to $\mathcal{A}$, an adversary against Σ. When $\mathcal{A}$ asks for signing query on m_i, $\mathcal{B}$ queries its own oracle on $m_i \| s_i$, where s_i is a random number of length ℓ. On receiving the signature σ, where $\sigma = \mathsf{S}'(sk, m\|s)$, $\mathcal{B}$ gives $\mathcal{A}$ $\sigma \oplus h(s)$ as corresponding answer. It is easily verified this is a valid signature of scheme Σ for $\mathcal{A}$.

Suppose that any valid forgery an adversary $\mathcal{A}$ outputs is of the form (m^*, s^*, σ^*), where $\sigma^* = \sigma'^* \oplus h(s^*)$ and $\sigma'^* = \mathsf{S}'(sk, m^*\|s^*)$. For a successful forgery, assume that any of (m^*, s^*, σ^*) is new, which implies that either (i) $m^*\|s^*$ is new; or (ii) $m^*\|s^*$ has been previously queried to the signing oracle, so there must be σ^* is new.

Case (i): $\mathcal{B}$ can extract σ'^* from σ by letting $\sigma' = \sigma \oplus (s^*)$. Thus $(m^*\|s^*, \sigma^* \oplus h(s^*))$ is a successful forgery for Σ', and this clearly contradicts the unforgeability of Σ.

Case (ii): Assume $\mathcal{A}$ asks at most q_s signing queries. Without loss of generality, we denote $m^* = m_j$ and $s^* = s_j$ for some $1 \leq j \leq q_s$. Notice that $\sigma^* \oplus h(s_j) \notin \{\sigma_i \oplus h(s_i)\}_{q_s}$, since $\sigma^* \neq \sigma_i$ for $1 \leq i \leq q_s$. However, according to the strong unforgeability of Σ', this can happen with only negligible probability which contradicts $\mathcal{A}$ is a successful forger for Σ.

Summarizing the above discussions, Theorem 3 is proved. □

5 Applications and Extensions

5.1 A Formal Proof for a Previous Scheme

Yang et al. [18] present a modification to their anonymous signature scheme when used in an anonymous paper review system. Besides signing on a message m, hash m with a random number s with a cryptographic hash function $H(\cdot)$ and sign on $H(m\|s)$. We are not aware of any formal analysis of this construction. For the paper review system, actually, weak anonymity is not sufficient. From Lemma 3 and Theorem 1, one immediately concludes that this construction is secure provided that H is a random oracle.

5.2 Easy and Efficient Instantiations

All our constructions are efficient, with performance comparable to the underlying non-anonymous signatures. It is easy to instantiate our generic constructions with practical signature schemes with a cryptographical pseudorandom generator. For example, it is easy to have efficiently strong anonymous signatures without assuming random oracles (cf. [18]), with underlying signature schemes also in the standard model [10, 3, 17, 4].

6 Conclusion

In this paper, we propose a new model of anonymous signatures, called strong anonymous signatures. The previous security definition grantees anonymity only if the message remains secret to the adversary, while in many applications the message actually has to be public. Our new definition removes this limitation by allowing the distinguisher to choose the challenge message.

Surprisingly, we show one can have this stronger security almost for free. We demonstrate several efficient constructions that satisfy our new definition, with the same set of assumptions as previous constructions. Some of our constructions assume no random oracles. Our results have some useful applications, e.g., a formal analysis of a previously unproven construction (in the random oracle model). Our results imply that *practical* anonymous signatures are actually easy to construct and easy to use.

References

1. J.H. An, Y. Dodis, and T. Rabin. On the Security of Joint Signature and Encryption. In *Eurocrypt'02*, volume 2332 of *LNCS*, pages 83–107. Springer, 2002.
2. M. Bellare, A. Boldyreva, A. Desai, and D. Pointcheval. Key-Privacy in Public-Key Encryption. In *Asiacrypt'01*, volume 2248 of *LNCS*, pages 566–582. Springer, 2001.
3. D. Boneh and X. Boyen. Short Sigantures without Random Oracles. In *EUROCRYPT'04*, volume 3027 of *LNCS*, pages 56–73. Springer, 2004.
4. D. Boneh, E. Shen, and B. Waters. Strongly Unforgeable Signatures Based on Computational Diffie-Hellman. In *PKC'06*, volume 3958 of *LNCS*, pages 229–240. Springer, 2006.
5. R. Canetti, Y. Dodis, S. Halevi, E. Kushilevitz, and A. Sahai. Exposure-Resilient Functions and All-or-Nothing Transforms. In *EUROCRYPT*, volume 1807 of *LNCS*, pages 453–469. Springer, 2000.
6. D. Chaum. Blind Signatures for Untraceable Payments. In *Proc. of CRYPTO'82*, pages 199–203. Plenum, 1983.
7. D. Chaum. Designated Comfirmer Sigantures. In *EUROCRYPT'94*, volume 950 of *LNCS*, pages 86–91. Springer, 1994.
8. D. Chaum and H. van Antwerpen. Undeniable Signatures. In *Crypto'89*, volume 435 of *LNCS*, pages 212–216. Springer, 1990.
9. D. Chaum and E. van Heyst. Group Sigantures. In *Eurocrypt'91*, volume 547 of *LNCS*, pages 257–265. Springer, 1991.
10. R. Cramer and V. Shoup. Signature schemes based on the strong rsa assumption. In *ACM CCS'99*, pages 46–51. ACM Press, 1999.
11. M. Fischlin. Anonymous Signatures Made Easy. In *PKC'07*, volume 4450 of *LNCS*, pages 31–42. Springer, 2007.
12. S. Goldwasser, S. Micali, and R.L. Rivest. A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.
13. M. Jakobsson, K. Sako, and R. Impagliazzo. Designated Verifier Proofs and Their Applications. In *Eurocrypt'96*, volume 1070 of *LNCS*, pages 143–154. Springer, 1996.
14. R. Rivest, A. Shamir, and Y. Tauman. How to Leak a Secret. In *Asiacrypt'01*, volume 2248 of *LNCS*, pages 552–565. Springer, 2001.
15. V. Shoup. Sequences of Games: a Tool for Taming Complexity in Security Proofs. Manuscript, 2004.
16. R. Steinfeld, L. Bull, H. Wang, and J. Pieprzyk. Universal Designated-Verifier Signatures. In *ASIACRYPT'03*, volume 2894 of *LNCS*, pages 523–542. Springer, 2003.
17. B. Waters. Efficient Identity-Based Encryption Without Random Oracles. In *EUROCRYPT'05*, volume 3494 of *LNCS*, pages 114–127. Springer, 2005.
18. G. Yang, D. Wong, X. Deng, and H. Wang. Anonymous Signature Schemes. In *PKC'06*, volume 3958 of *LNCS*, pages 347–363. Springer, 2006.