Anonymous Signatures Revisited

Vishal Saraswat and Aaram Yun

University of Minnesota — Twin Cities
{vishal,aaram}@cs.umn.edu

Abstract. We revisit the notion of the anonymous signature, first formalized by Yang, Wong, Deng and Wang [7], and then further developed by Fischlin [5] and Zhang and Imai [8]. We point out that the previous formalism is inadequate in several aspects and present a new formalism. We introduce the notion *unpretendability* to guarantee infeasibility for someone other than the correct signer to pretend authorship of the message and signature. Our definition retains applicability for all previous applications of the anonymous signature, provides stronger security, and is conceptually simpler. We give a generic construction from any ordinary signature scheme and finally we present an example construction of an efficient anonymous signature scheme. We show that the short signature scheme by Boneh and Boyen [3] can be naturally regarded as such a secure anonymous signature scheme according to our formalism.

Keywords: anonymous signature, signature, anonymity, unpretendability

1 Introduction

An anonymous signature is a signature scheme where the signature σ of a message m does not reveal the identity of the signer. Yang et al. [7] discussed the usefulness of anonymous signatures in many applications where anonymity is needed, including key exchange protocols, auction systems, and annonymous paper reviewing.

The notion of the anonymous signature was formalized much later than that of the anonymous encryption. Bellare et al. [1] had already defined in Asiacrypt 2001 key-privacy, or anonymity of an encryption scheme, as indistinguishability of ciphertexts encrypted by different public keys, that is, an eavesdropper cannot obtain any information about the recipient (corresponding to the public key) from the ciphertext. In a way, this delay is not too surprising. One obvious problem for introducing the idea of anonymity to digital signatures is that a signature is publically verifiable; if there are only a few candidate signers, the adversary of anonymity can simply try verification of the message-signature pair with respect to all candidate public keys to break anonymity. Therefore, as long as the adversary obtains both the message and the signature, it seems that anonymity is impossible.

Yang et al. solved the paradox by guaranteeing the anonymity only when the adversary obtains the signature and not the message, or when there is some randomness in the message not revealed to the adversary. While the idea of revealing only the signature and not the message sounds strange at first, actually in many applications it makes sense; for example, in the key transport example given by Yang et al., Bob already knows what Alice's message should be from previous communication, so Alice may send only the anonymous signature without the message, and this authenticates Alice while protecting Alice's anonymity from eavesdroppers. In the case of an auction, a bidder may append some random string r to a message m, which is his bid, and sign it. After the auction ends, only the winner may reveal the randomness r and thus his identity, and the other participants remain anonymous.

This idea of hidden randomness in the message is used by Fischlin [5] to propose a generic transformation for anonymous signatures out of ordinary signatures, by applying the idea of randomness extractor to extract the hidden randomness and use it for anonymizing the signature. Fischlin's formulation of anonymous signatures is slightly different, but essentially captures the same idea as that of Yang et al. Also in [8], Zhang and Imai suggested the notion of 'strong anonymous signatures', where they considered the case when there is not much uncertainty in the message.

1.1 Limits of the previous formalism

We revisit the formal definition of anonymous signature and show that previous formalisms of anonymous signature are not adequate in that, they fail to capture the intuition fully, are not completely usable for suggested applications, and actually are inconsistent with what happens for those applications. Also, we claim that a slightly different formalism captures the intuition better, retains the applicability, more consistently models the application scenarios, enables simpler constructions, and gives better security guarantee.

As explained, in the current formalism, the signer anonymity is based on hidden residual randomness of the message. As long as there is enough such randomness, the signer maintains anonymity, but of course the signature cannot be verified. Eventually the randomness in message is revealed explicitly or implicitly, and whoever has the complete message-signature pair can verify the signature.

In order to model this, Yang et al. and Fischlin formalize that each signer, having public key pk, has certain message distribution $\mathcal{M}(pk)$. Then, two key pairs (pk_0, sk_0), (pk_1, sk_1) are chosen and pk_0 and pk_1 are given to the adversary. Also, a message m is chosen from $\mathcal{M}(pk_b)$ with respect to a random bit $b \in \{0, 1\}$, and the signature $\sigma = \mathrm{Sig}(sk_b, m)$ is computed and given to the adversary. If the adversary cannot guess the random bit b with probability not much greater than $1/2$, then the signature scheme is considered anonymous.

But this formalism is not satisfactory in several aspects. First, this is in fact *inconsistent* to the suggested application of anonymous auction, or anonymous paper review. In these cases, if m is the original intended message, then the

signer adds some random string r to form appended message $m\|r$, and releases the message m, together with the signature σ of the appended message $m\|r$. From the point of view of an eavesdropper, different original message m gives different message distribution of the whole appended message $m\|r$; the message distribution cannot be a function of only the public key pk, and in fact also depends on the partially revealed portion (m) of the message.

Second, this definition does not formally give a guarantee of infeasibility for someone other than the correct signer to come later and pretend that the signature is his. We call this property *unpretendability*. For an ordinary signature for which complete message-signature pair is released at once, this problem does not arise; the pair is publically verifiable and the authorship can be attributed to the signer. But for an anonymous signature, where only a part of the message-signature pair is released initially, there is theoretical possibility that someone other than the signer may come and claim the authorship of the message and signature. For example, in the anonymous paper review example, the author A of a paper $paper_A$ picks a random string r, computes $\sigma \leftarrow \mathrm{Sig}(sk_A, paper_A\|r)$, and releases $(paper_A, \sigma)$ initially, and only later reveals r when the paper is accepted. Now, if the anonymous signature is not unpretendable, then another author, B, may be able to compute r' satisfying $\mathrm{Vf}(pk_B, paper_A\|r', \sigma) = \mathsf{true}$ and use such an r' to claim authorship of $paper_A$.

Hence, we argue that this unpretendability should be an essential feature of an anonymous signature; otherwise anonymous signature is in fact not applicable for quite a few of originally proposed applications.

Note that we are not claiming that any of the actual schemes proposed in previous papers fails to satisfy unpretendability. But, still this notion should be formally defined and guaranteed for each anonymous scheme. In fact, later we will give an example of an unforgeable signature scheme which provides complete anonymity but is not at all unpretendable. This means that, unpretendability does not follow directly from unforgeability and/or anonymity, and warrants separate definition.

Third, we feel that the idea of a signature of an unknown message counter-intuitive. Intuitively, a signature is a proof of authorship for a given document. If we do not know the document in question, or if we are not sure whether the document ends with 'Therefore you should ...,' or 'Therefore you should not ...,' then the meaning of a signature for such uncertain document is at least debatable.

1.2 Our formalism

Discarding hidden randomness in the message. For these reasons, we believe that the previous formalism is inadequate, and we fix the definition as follows: first, instead of relying on the hidden residual randomness of the message, we use an explicitly given randomness. Second, we formalize not only the notion of anonymity, but also give explicit formalization of unpretendability.

In traditional digital signatures, signature generation is considered as a randomized algorithm in general, therefore this strategy of explicit randomness is

applicable no matter how much entropy (or lack thereof) the distribution of the message has.

This enables us to discard the randomness extraction from the message altogether, and use the provided randomness directly to anonymize the public key. In fact, even when there is enough entropy in the message distribution, often the randomness is not diffused in the whole message but well-separated from the rest of the message and controllable by the signer. For example, in the bidding example where the bidder appends some random string r to the message m and then sign the appended message $m\|r$, certainly the distribution of this appended message has enough entropy which can be extracted back, but we feel this is artificial; the original message was m, and intuitively, the signer is not really interested in protecting the integrity of r, which is not part of his message m which he *really* wanted to sign. Hence, it is more natural to regard this r as a separate parameter involved in the signature generation and which is part of the signature, instead of artificially regarding this as a part of the message which needs to be signed and protected.

Surfacing the verification token. Therefore, in our formalism, we split a digital signature $\tilde{\sigma}$ into two parts, $\tilde{\sigma} = (\sigma, \tau)$. We call τ a *verification token*, or a token in short, which is chosen uniform randomly from a large space. Then σ, the rest of $\tilde{\sigma}$, is now just called a *signature*. The signature σ is computed by the signature generation algorithm which takes the signer's secret key, the message m and the token τ as inputs, and when m, σ, and τ are presented, then anyone can verify the validity of the signature using the public key of the signer. But as long as τ is hidden, the adversary cannot break the anonymity of the signer just from the message m and the signature σ. Meanwhile, anyone to whom the token τ (along with the identity of the signer) is revealed may verify the signature.

Note that our formalism is just a specialization of the traditional formalism of digital signature, and not something incompatible; (σ, τ) together serves as a signature which is publically verifiable, and unforgeable according to the usual definition. We only enforce our signature to have this special format, and to have anonymity and unpretendability in addition to the unforgeability.

In short, we surfaced the hidden randomness of the anonymous signature explicit as the verification token, and moved it from the message to the signature itself. Also we identified and formalized the unpretendability as another property an anonymous signature should have.

Regarding randomness extraction. In Fischlin's general transformation, he uses a randomness extractor to extract randomness from the message, and anonymizes the signature using this extracted randomness. In our formalism, this extracted randomness serves the role of the verification token. We believe that it is better to regard this randomness extraction as external to the anonymous signature, because we feel that how the verification token is generated is not essential to the function of the anonymous signature. Depending on the application, if it is necessary one may use randomness extractor to generate the token. Or in many other cases, one may rely on more straightforward methods. By separating the

randomness extraction out of the signature, we obtain a conceptually simpler and more efficient formalism.

Enhanced notion of security. Not only separating the randomness extraction from the anonymous signature results in a conceptually cleaner formalism, but also it enables us to guarantee better notion of security. Because in previous formalisms the verification token was 'diffused' in the message itself, the adversary of anonymity could not choose the challenge message by himself, and a random challenge message had to be chosen out of some message distribution. But in our formalism, there is no problem for the adversary to adaptively choose the challenge message by himself, and indeed we give this stronger notion of anonymity, which all of our schemes meet.

Our contribution. Therefore, in this paper, we give a new formalism for an anonymous signature following the outline given in the introduction. We re-examine the suggested applications for anonymous signatures, and show that our new formalism retains applicability. Also, we present some construction of efficient anonymous signature schemes. We first give a generic construction out of any ordinary unforgeable signature scheme. Also, we show that the short signature scheme by Boneh and Boyen [3] can be naturally regarded as such a secure anonymous signature scheme according to our formalism with essentially no modification.

2 Related work

The notion of anonymous signature was first formalized by Yang et al. in [7], and explored further by Fischlin in [5]. Our work revisits this notion, and provides an alternative formalism which fixes some inadequacies of the previous formalism.

Zhang and Imai [8] proposed a very similar approach as ours. Their idea is to define 'strong anonymous signature', which maintains anonymity even when there is not much uncertainty in the message distribution. In fact, their definition of strong anonymity is essentially identical to our anonymity. We remark that we had obtained our results independently before we learned about their work. In comparison with our formalism, they define their strong anonymity as a stronger version of anonymity than the original definition, while we discard the previous definition as inadequate and reformulate anonymity. Also, Zhang and Imai do not discuss unpretendability, which we argue as central to the notion of anonymous signatures.

Galbraith and Mao [6] introduced the notion of anonymity to undeniable and confirmer signatures. Our definition of anonymity of an anonymous signature is similar to theirs, and also the fact that the signer has to provide the verification token later to let others verify the signature looks similar to the case of undeniable signatures. But we stress that an anonymous signature is not an undeniable signature; anyone who obtained the token of the signature can in fact let others verify the signature, without involvement of the signer. In general, an anonymous signature is much simpler than an anonymous undeniable signature.

3 Definition

3.1 Notations and conventions

We denote by $v \leftarrow A(x, y, z, \ldots)$ the operation of running a randomized or deterministic algorithm $A(x, y, z, \ldots)$ and storing the output to the variable v. If X is a set, then $v \xleftarrow{R} X$ denotes the operation of choosing an element v of X according to the uniform random distribution on X. Unless stated otherwise, all algorithms are probabilistic.

3.2 Anonymous signature

We define an *anonymous signature* Σ as a quadruple of polynomial-time algorithms $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$, where the parameter generation algorithm Par() outputs a common parameter $P \leftarrow \mathrm{Par}(1^k)$ using security parameter k, the key generation algorithm Gen() outputs a key pair $(pk, sk) \leftarrow \mathrm{Gen}(P)$ given the common parameter P as input, signature generation algorithm Sig() outputs a signature $\sigma \leftarrow \mathrm{Sig}(sk, m, \tau)$ with respect to the secret key sk, a message $m \in \{0,1\}^*$ and a *verification token* $\tau \xleftarrow{R} \mathcal{T}(k)$ ($\mathcal{T}(k)$ is the space of verification tokens for the security parameter k), and the deterministic, signature verification algorithm $\mathrm{Vf}(pk, m, \sigma, \tau)$ outputs true or false.

The common parameter P contains the security parameter k itself, and additionally contain other public information common to all users of the system, for example description of cryptographic groups used in the signature scheme.

For consistency, we require the following:

$$\mathrm{Vf}(pk, m, \mathrm{Sig}(sk, m, \tau), \tau) = \mathsf{true},$$

for $P \leftarrow \mathrm{Par}(1^k)$, $(pk, sk) \leftarrow \mathrm{Gen}(P)$, $\tau \xleftarrow{R} \mathcal{T}(k)$, and for any $m \in \{0,1\}^*$.

3.3 Unforgeability

We say that $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$ is *unforgeable*, if for any polynomial-time adversary $\mathcal{A}$, the advantage $\mathbf{Adv}^{\mathsf{uf\text{-}cma}}_{\Sigma,\mathcal{A}}(k)$ which is defined as

$$\mathbf{Adv}^{\mathsf{uf\text{-}cma}}_{\Sigma,\mathcal{A}}(k) \stackrel{\mathrm{def}}{=} \mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{uf\text{-}cma}}_{\Sigma,\mathcal{A}}(k) = \mathsf{true}\right]$$

is negligible in the following experiment:

Experiment $\mathbf{Expr}^{\mathsf{uf\text{-}cma}}_{\Sigma,\mathcal{A}}(k)$
  $P \leftarrow \mathrm{Par}(1^k)$
  $(pk, sk) \leftarrow \mathrm{Gen}(P)$
  $(m^*, \sigma^*, \tau^*) \leftarrow \mathcal{A}^{\mathrm{Sig}(sk,\cdot,\cdot)}(pk)$
  return $\mathrm{Vf}(pk, m^*, \sigma^*, \tau^*)$

where the adversary $\mathcal{A}$ has access to the signing oracle $\mathrm{Sig}(sk, \cdot, \cdot)$ with respect to the secret key sk. We also require that $\mathcal{A}$ is not allowed to query the signing oracle with (m^*, τ) for any τ.

Similarly, we say that Σ is *strongly unforgeable*, if for any polynomial-time adversary $\mathcal{A}$, the advantage $\mathbf{Adv}^{\mathsf{suf\text{-}cma}}_{\Sigma,\mathcal{A}}(k)$ which is defined as

$$\mathbf{Adv}^{\mathsf{suf\text{-}cma}}_{\Sigma,\mathcal{A}}(k) \stackrel{\text{def}}{=} \mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{suf\text{-}cma}}_{\Sigma,\mathcal{A}}(k) = \mathsf{true}\right]$$

is negligible in the following experiment:

$$\begin{array}{l}\text{Experiment } \mathbf{Expr}^{\mathsf{suf\text{-}cma}}_{\Sigma,\mathcal{A}}(k)\\ \quad P \leftarrow \mathrm{Par}(1^k)\\ \quad (pk, sk) \leftarrow \mathrm{Gen}(P)\\ \quad (m^*, \sigma^*, \tau^*) \leftarrow \mathcal{A}^{\mathrm{Sig}(sk,\cdot,\cdot)}(pk)\\ \quad \textbf{return } \mathrm{Vf}(pk, m^*, \sigma^*, \tau^*)\end{array}$$

The experiment looks identical to the above, but the difference is that, for this, we require $\mathcal{A}$ not to have received σ^* as an answer to any query of form (m^*, τ^*) to the signing oracle.

Remark 1. In our definition of unforgeability and other properties, we allow the adversary to have a signing oracle of form $\mathrm{Sig}(sk, \cdot, \cdot)$, i.e., we allow the adversary to choose the verification token, too. Since the token is not a part of the message and is selected internally by the signer, disallowing this would be also reasonable. Note that there is no extra cost in achieving the stronger definition.

3.4 Anonymity

Consider an adversary which is a pair of polynomial-time algorithms $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$. Let st be the state information which $\mathcal{A}_1$ passes to $\mathcal{A}_2$. We say that $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$ is *anonymous*, if for any such $\mathcal{A}$, the advantage $\mathbf{Adv}^{\mathsf{anon}}_{\Sigma,\mathcal{A}}(k)$ defined as

$$\mathbf{Adv}^{\mathsf{anon}}_{\Sigma,\mathcal{A}}(k) \stackrel{\text{def}}{=} \left|\mathbf{Pr}[\mathbf{Expr}^{\mathsf{anon\text{-}1}}_{\Sigma,\mathcal{A}}(k) = 1] - \mathbf{Pr}[\mathbf{Expr}^{\mathsf{anon\text{-}0}}_{\Sigma,\mathcal{A}}(k) = 1]\right|$$

is negligible, where experiments $\mathbf{Expr}^{\mathsf{anon}\text{-}b}_{\Sigma,\mathcal{A}}$ $(b = 0, 1)$ are defined as follows:

$$\begin{array}{l}\text{Experiment } \mathbf{Expr}^{\mathsf{anon}\text{-}b}_{\Sigma,\mathcal{A}}(k)\\ \quad P \leftarrow \mathrm{Par}(1^k)\\ \quad (pk_0, sk_0) \leftarrow \mathrm{Gen}(P);\ (pk_1, sk_1) \leftarrow \mathrm{Gen}(P)\\ \quad (m^*, st) \leftarrow \mathcal{A}_1^{\mathrm{Sig}(sk_0,\cdot,\cdot),\mathrm{Sig}(sk_1,\cdot,\cdot)}(pk_0, pk_1)\\ \quad \tau^* \stackrel{\mathrm{R}}{\leftarrow} \mathcal{T}(k)\\ \quad \sigma^* \leftarrow \mathrm{Sig}(sk_b, m^*, \tau^*)\\ \quad b' \leftarrow \mathcal{A}_2^{\mathrm{Sig}(sk_0,\cdot,\cdot),\mathrm{Sig}(sk_1,\cdot,\cdot)}(\sigma^*, st)\\ \quad \textbf{return } b'\end{array}$$

We call Σ anonymous with respect to *full key exposure*, when the advantage of any adversary is still negligible even if the adversary also gets the secret keys sk_0, sk_1 as additional input. We denote by $\mathbf{Adv}^{\mathsf{anon\text{-}fke}}_{\Sigma,\mathcal{A}}(k)$ the advantage of an adversary in the anonymity experiment with full key exposure.

3.5 Unpretendability

We say that $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$ is *unpretendable*, if for any adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, the advantage $\mathbf{Adv}^{\mathsf{up}}_{\Sigma,\mathcal{A}}(k)$ defined as

$$\mathbf{Adv}^{\mathsf{up}}_{\Sigma,\mathcal{A}}(k) \stackrel{\mathrm{def}}{=} \mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{up}}_{\Sigma,\mathcal{A}}(k) = \mathsf{true}\right]$$

is negligible in the following experiment:

$$\begin{array}{l}
\text{Experiment } \mathbf{Expr}^{\mathsf{up}}_{\Sigma,\mathcal{A}}(k) \\
\quad P \leftarrow \mathrm{Par}(1^k) \\
\quad (pk, sk) \leftarrow \mathrm{Gen}(P);\ (pk^*, sk^*) \leftarrow \mathrm{Gen}(P) \\
\quad (m^*, st) \leftarrow \mathcal{A}_1^{\mathrm{Sig}(sk^*,\cdot,\cdot)}(pk^*, pk, sk) \\
\quad \tau^* \stackrel{\mathrm{R}}{\leftarrow} \mathcal{T}(k) \\
\quad \sigma^* \leftarrow \mathrm{Sig}(sk^*, m^*, \tau^*) \\
\quad \tau \leftarrow \mathcal{A}_2^{\mathrm{Sig}(sk^*,\cdot,\cdot)}(\sigma^*, \tau^*, st) \\
\quad \textbf{return } \mathrm{Vf}(pk, m^*, \sigma^*, \tau)
\end{array}$$

Intuitively, the adversary's key pair is (pk, sk), and he is trying to claim the authorship of (m^*, σ^*), which is signed by the target secret key sk^* with the verification token τ^*. The adversary tries to produce an appropriate τ satisfying $\mathrm{Vf}(pk, m^*, \sigma^*, \tau) = \mathsf{true}$, and the definition guarantees that the success probability for this attempt is negligible.

Like the case of anonymity, we say that Σ is unpretendable with respect to full key exposure, when the advantage of any adversary is still negligible even if the adversary also gets the target secret key sk^* as additional input. We denote by $\mathbf{Adv}^{\mathsf{up\text{-}fke}}_{\Sigma,\mathcal{A}}(k)$, the advantage of an adversary in the unpretendability experiment with full key exposure.

3.6 Security of an anonymous signature

Suppose that $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$ is an anonymous signature scheme. We say that Σ is a *secure* anonymous signature, if Σ is unforgeable, anonymous, and unpretendable.

We emphasize that the unpretendability is a crucial property that an anonymous signature should have. Already we showed that if an anonymous signature is not unpretendable, then it cannot be used for some of the suggested applications like anonymous paper review. Here, let us show an example of an anonymous signature which is unforgeable, anonymous, but not unpretendable.

Suppose $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$ is an ordinary unforgeable signature scheme. We then construct an anonymous signature scheme $\Sigma' = (\mathrm{Par}', \mathrm{Gen}', \mathrm{Sig}', \mathrm{Vf}')$ as follows: $\mathrm{Par}'(1^k)$ is the same as $\mathrm{Par}(1^k)$, $\mathrm{Gen}'(P)$ is the same as $\mathrm{Gen}(P)$. $\mathrm{Sig}'(sk, m, \tau)$ is defined as

$$\mathrm{Sig}'(sk, m, \tau) \stackrel{\mathrm{def}}{=} \mathrm{Sig}(sk, m) \oplus \tau$$

where the verification token τ is chosen to be of the same bit-length as the signature $\mathrm{Sig}(sk, m)$. Finally, $\mathrm{Vf}'(sk, m, \sigma, \tau)$ is defined as

$$\mathrm{Vf}'(pk, m, \sigma, \tau) \stackrel{\text{def}}{=} \mathrm{Vf}(pk, m, \sigma \oplus \tau).$$

It is clear that the anonymous signature Σ' is both unforgeable and anonymous; because the signature $\sigma = \mathrm{Sig}(sk, m)$ is masked with random bitstring τ in $\mathrm{Sig}'(sk, m, \tau)$, essentially the adversary has no information about the signature. Only when later τ is revealed, the signature σ is revealed and signature can be verified. Thus, this is equivalent to defering the signing to the last minute when the token τ has to be revealed. Hence the scheme is unforgeable, and unless τ is revealed, the signer anonymity is guaranteed.

But, it is trivial to break unpretendability of this scheme; if $(m^*, \sigma^* = \mathrm{Sig}(sk^*, m^*) \oplus \tau^*)$ is given, then the adversary may compute $\mathrm{Sig}(sk, m^*)$ using his own secret key sk, and compute τ as

$$\tau \stackrel{\text{def}}{=} \mathrm{Sig}(sk, m^*) \oplus \sigma^*.$$

Then,

$$\mathrm{Vf}'(pk, m^*, \sigma^*, \tau) = \mathrm{Vf}(pk, m^*, \sigma^* \oplus \tau) = \mathrm{Vf}(pk, m^*, \mathrm{Sig}(sk, m^*)) = \mathsf{true}.$$

Also, while we present this example in our formalism, it is also possible to modify this example to fit the previous formalism.

4 Applications

We note that our new formalism still allows an anonymous signature scheme to be applicable for all areas suggested for original formalism. For anonymous auction and anonymous paper review, if m is the message to be signed, then the signer can choose a verification token $\tau \stackrel{\text{R}}{\leftarrow} \mathcal{T}$ privately, compute $\sigma \leftarrow \mathrm{Sig}(sk, m, \tau)$, and release (m, σ). As long as τ is not revealed, the anonymity is preserved. Only when necessary, the signer can reveal τ and claim the authorship of (m, σ). In fact, our formalism guarantees that this authorship can be claimed without false pretension from other person, because of the unpretendability of the anonymous signature scheme.

For some other applications, the signer does not in fact reveal the verification token explicitly, because the other party already knows it according to some outer protocol. Consider the example of authenticated key transport protocol given by Yang et al. [7], which was originally proposed by Boyd and Park [4].

$$\begin{aligned}
&A \rightarrow B : PKE_B(ID_A, k, count),\\
&A \leftarrow B : Enc_k(count, r_B),\\
&A \rightarrow B : \mathrm{Sig}_A(ID_B, h(count, k, r_B)).
\end{aligned}$$

In this protocol, an mobile client A wants to transport the symmetric key k to a server B, securely and while protecting the anonymity of A. In our formalism of anonymous signatures, we may modify the last line as

$$A \to B : \mathrm{Sig}(sk_A, ID_B \| count \| r_B, k).$$

Here A does not have to reveal the verification token $\tau = k$ later to B, because A already sent k to B on the first line. Note that here we are not to present a provably secure key transport scheme but to demonstrate the applicability of our formalism.

5 Secure anonymous signature schemes

In this section, we exhibit a few anonymous signature schemes. First, we show how to construct an anonymous signature scheme generically from any ordinary unforgeable signature scheme. Then, we show that the short signature scheme of Boneh and Boyen [3] can be naturally considered as a secure anonymous signature according to our formalism, with essentially no modification.

5.1 Generic construction

Here we present a generic construction of an anonymous signature scheme using an ordinary signature scheme, a one-way hash function, and a pseudorandom generator secure with respect to the hash function. For the signature scheme, it is required that it is unforgeable, and the public key size and the signature size are constant for all users (for any security parameter k).

Let $\Sigma = (\mathrm{Par}, \mathrm{Gen}, \mathrm{Sig}, \mathrm{Vf})$ be a signature scheme. When k is the security parameter, let $l_p(k)$ and $l_s(k)$ be the bit length of a public key pk and the bit length of a signature, respectively. We need a collision-resistant function $H : \{0,1\}^{l_0(k)} \to \{0,1\}^{l_1(k)}$, and a pseudorandom generator $G : \{0,1\}^{l_0(k)} \to \{0,1\}^{l_p(k)+l_s(k)}$, where $l_0(k)$, $l_1(k)$, $l_p(k)$, and $l_s(k)$ are some polynomial functions of k. We construct an anonymous signature $\Sigma' = (\mathrm{Par}', \mathrm{Gen}', \mathrm{Sig}', \mathrm{Vf}')$ using these as follows:

function $\mathrm{Par}'(1^k)$
  $P \leftarrow \mathrm{Par}(1^k)$
  $P' \leftarrow P \| G \| H$
  return P'

function $\mathrm{Gen}'(P')$
  Parse P' as $P \| G \| H$
  $(pk, sk) \leftarrow \mathrm{Gen}(P)$
  $pk' \leftarrow pk \| G \| H$
  $sk' \leftarrow sk \| pk \| G \| H$
  return (pk', sk')

function $\mathrm{Sig}'(sk', m, \tau)$
  Parse sk' as $sk \| pk \| G \| H$
  return $((pk \| \mathrm{Sig}(sk, m \| H(\tau))) \oplus G(\tau)) \| H(\tau)$

function $\mathrm{Vf}'(pk', m, \sigma, \tau)$
  Parse pk' as $pk \| G \| H$
  Parse σ as $\sigma_1 \| \sigma_2$
  if $\sigma_2 \neq H(\tau)$ **then**
    return false
  Parse $\sigma_1 \oplus G(\tau)$ as $\sigma_3 \| \sigma_4$
  return $(\sigma_3 = pk) \wedge \mathrm{Vf}(pk, m \| \sigma_2, \sigma_4)$

The properties we require of H and G are as follows. First, for the function H, we want it to be collision resistant: for any polynomial-time adversary $\mathcal{A}$, the advantage $\mathbf{Adv}^{\mathsf{cr}}_{H,\mathcal{A}}(k)$ defined as

$$\mathbf{Adv}^{\mathsf{cr}}_{H,\mathcal{A}}(k) \stackrel{\text{def}}{=} \mathbf{Pr}[\mathbf{Expr}^{\mathsf{cr}}_{H,\mathcal{A}}(k) = \mathsf{true}]$$

is negligible in the following experiment:

Experiment $\mathbf{Expr}^{\mathsf{cr}}_{H,\mathcal{A}}(k)$
  $(\tau, \tau') \leftarrow \mathcal{A}(1^k)$
  if $\tau \notin \{0,1\}^{l_0(k)}$ or $\tau' \notin \{0,1\}^{l_0(k)}$ **then**
    return false
  return $\tau' \neq \tau \wedge H(\tau') = H(\tau)$

Note that we do not call H a hash function, because we do not require H to compress; the domain of H does not have to be larger than the codomain.

For the pseudorandom generator G, we want $G(\tau)$ to be pseudorandom even when $H(\tau)$ is exposed. More precisely, for any polynomial-time adversary $\mathcal{A}$, the advantage $\mathbf{Adv}^{\mathsf{prg}}_{G,H,\mathcal{A}}(k)$ defined as

$$\mathbf{Adv}^{\mathsf{prg}}_{G,H,\mathcal{A}}(k) \stackrel{\text{def}}{=} \left|\mathbf{Pr}[\mathbf{Expr}^{\mathsf{prg\text{-}1}}_{G,H,\mathcal{A}}(k) = 1] - \mathbf{Pr}[\mathbf{Expr}^{\mathsf{prg\text{-}0}}_{G,H,\mathcal{A}}(k) = 1]\right|$$

is negligible, where experiments $\mathbf{Expr}^{\mathsf{prg}\text{-}b}_{G,H,\mathcal{A}}$ $(b = 0, 1)$ are defined as follows:

Experiment $\mathbf{Expr}^{\mathsf{prg}\text{-}b}_{G,H,\mathcal{A}}(k)$
  $\tau \stackrel{\text{R}}{\leftarrow} \{0,1\}^{l_0(k)}$
  $R_0 \stackrel{\text{R}}{\leftarrow} \{0,1\}^{l_p(k)+l_s(k)}$
  $R_1 \leftarrow G(\tau)$
  $b' \leftarrow \mathcal{A}(1^k, R_b, H(\tau))$
  return b'

When G satisfies the above, we call G to be a pseudorandom generator with respect to H.

It is not difficult to instantiate the pair G and H satisfying the above. Indeed, it can be trivially instantiated in the random oracle model, and in the standard model, we may construct one from a one-way permutation: given a one-way permutation $\pi : \{0,1\}^{l_0(k)} \to \{0,1\}^{l_0(k)}$ and its hard-core bit $b : \{0,1\}^{l_0(k)} \to \{0,1\}$, we can use the standard Blum-Micali construction [2] based on hard-core bits:

$$G(\tau) \stackrel{\text{def}}{=} b(\pi^L(\tau))\|b(\pi^{L-1}(\tau))\|\cdots\|b(\pi^2(\tau))\|b(\pi(\tau)),$$

where $L = l_p(k) + l_s(k)$. H can then simply be defined as $H(\tau) \stackrel{\text{def}}{=} \pi^{L+1}(\tau)$. It is well known that $G(\tau)\|H(\tau)$ itself is computationally indistinguishable from a uniformly random bitstring, so G is a pseudorandom generator with respect to H, and because H is a one-way permutation, it is trivially collision resistant.

Using stronger assumptions, we may instantiate G and H more efficiently. For example, one can use decisional Diffie-Hellman assumption or its hashed

variants to construct G and H: let $\mathbb{G}$ be a cyclic group of prime order, and let g be a random generator of $\mathbb{G}$ and h a random element of $\mathbb{G}$. If the decisional Diffie-Hellman assumption holds, then (g, h, g^r, h^r) and (g, h, g^r, k) for a uniformly and independently chosen $k \stackrel{\mathrm{R}}{\leftarrow} \mathbb{G}$ are indistinguishable. Then $G(r) \stackrel{\mathrm{def}}{=} h^r$, $H(r) \stackrel{\mathrm{def}}{=} g^r$ satisfies the required properties.

Remark 2. The above construction is aimed at preserving not only unforgeability but also strong unforgeability. If we are interested only in preserving unforgeability, then we may simplify the construction a little: $\mathrm{Sig}'(sk', m, \tau)$ can be defined as

$$\mathrm{Sig}'(sk', m, \tau) \stackrel{\mathrm{def}}{=} ((pk \| \mathrm{Sig}(sk, m)) \oplus G(\tau)) \| H(\tau)$$

Remark 3. Our generic construction is similar to the construction given by Zhang and Imai in Section 4.2 of their paper [8]. We note that care is needed for that construction: in our notation, they defined $\mathrm{Sig}'(sk', m, \tau)$ to be $\mathrm{Sig}(sk, m\|\tau) \oplus G(\tau)$. In their construction, it is not sufficient for G to be a pseudorandom generator. This is because $\mathrm{Sig}(sk, m\|\tau)$ and $G(\tau)$ are correlated by the hidden variable τ. In order to prove anonymity of this construction, G has to look pseudorandom even when $\mathrm{Sig}(sk, m\|\tau)$ is exposed: for example, suppose we are given an unforgeable signature $\overline{\mathrm{Sig}}()$. Using this, we construct $\mathrm{Sig}(sk, m\|\tau) \stackrel{\mathrm{def}}{=} \overline{\mathrm{Sig}}(sk, m\|\tau) \oplus G(\tau)$, i.e., in order to sign a message with length larger than or equal to $l_0(k)$, sign the message and xor it with the output of the pseudorandom generator for the last $l_0(k)$ bits of the message. In that case, the construction of Zhang and Imai gives $\mathrm{Sig}'(sk', m, \tau) = \mathrm{Sig}(sk, m\|\tau) \oplus G(\tau) = \overline{\mathrm{Sig}}(sk, m\|\tau)$. If $\overline{\mathrm{Sig}}$ leaks information about pk corresponding to sk, then so does Sig'.

Note that in contrast to our construction, they allow G to be different between different users, so this example is not directly applicable. But still G has to be a pseudorandom generator satisfying the stronger property.

5.2 Security of the generic construction

Theorem 1. *Given an ordinary signature scheme Σ, consider the scheme Σ' defined in the previous subsection. If Σ is unforgeable, then Σ' is a secure unforgeable anonymous signature. Moreover, Σ' is both anonymous and unpretendable with respect to full key exposure. Also, if Σ is strongly unforgeable, then Σ' is also a secure strongly unforgeable anonymous signature.*

Proof. We only give proof for the case when the underlying signature scheme Σ is strongly unforgeable, because the other case can be proved similarly.

First, let us prove the strong unforgeability of Σ'. Suppose that $\mathcal{A}$ is an adversary attacking strong unforgeability of Σ'. Then using $\mathcal{A}$, we construct an adversary $\mathcal{B}$ which attacks strong unforgeability of Σ, and an adversary $\mathcal{C}$ attacking collision resistance of H, and together satisfying

$$\mathbf{Adv}^{\mathsf{suf\text{-}cma}}_{\Sigma',\mathcal{A}}(k) \leq \mathbf{Adv}^{\mathsf{suf\text{-}cma}}_{\Sigma,\mathcal{B}}(k) + \mathbf{Adv}^{\mathsf{cr}}_{H,\mathcal{C}}(k).$$

The adversary $\mathcal{B}$ is given a public key pk of Σ, and the corresponding signing oracle $\mathrm{Sig}(sk, \cdot)$. $\mathcal{B}$ computes $pk' \leftarrow pk\|G\|H$, and gives it to $\mathcal{A}$. $\mathcal{B}$ keeps a list

L which is initialized to an empty set. And, $\mathcal{B}$ answers the signing query of $\mathcal{A}$ as follows: for signing query of (m, τ), $\mathcal{B}$ appends τ to the list L, and calls its own signing oracle with query $m\|H(\tau)$. When it obtains its answer σ, $\mathcal{B}$ returns $\sigma' = ((pk\|\sigma) \oplus G(\tau))\|H(\tau)$. Note that the simulation is perfectly done according to the description of Σ'. Suppose that $\mathcal{A}$ halts with output (m^*, σ^*, τ^*). Then $\mathcal{B}$ first checks if L contains any τ with $\tau \neq \tau^*$ and $H(\tau) = H(\tau^*)$. In that case, $\mathcal{B}$ halts with some arbitrary output. If L does not contain any such element, then $\mathcal{B}$ parses σ^* as $\sigma_1\|\sigma_2$, parses $\sigma_1 \oplus G(\tau^*)$ as $\sigma_3\|\sigma_4$, and halts with output $(m^*\|\sigma_2, \sigma_4)$.

Now, the description of $\mathcal{C}$ is almost identical to that of $\mathcal{B}$: $\mathcal{C}$ provides the same simulation for $\mathcal{A}$ as $\mathcal{B}$, up to the step where $\mathcal{A}$ halts with output (m^*, σ^*, τ^*). $\mathcal{C}$ also checks if L contains any τ with $\tau \neq \tau^*$ and $H(\tau) = H(\tau^*)$, and in that case, halts with output (τ, τ^*). Otherwise, L halts with some arbitrary output.

We show that, whenever the output (m^*, σ^*, τ^*) of $\mathcal{A}$ is a successful forgery for Σ', then either $\mathcal{B}$ finds an element $\tau \in L$ such that $\tau \neq \tau^*$ and $H(\tau) = H(\tau^*)$, or $(m^*\|\sigma_2, \sigma_4)$ is a successful forgery for Σ. Suppose that (m^*, σ^*, τ^*) is a successful forgery for Σ', and also that no $\tau \in L$ satisfies $\tau \neq \tau^*$ and $H(\tau) = H(\tau^*)$. From the definition of Vf', since $\mathrm{Vf}'(sk', m^*, \sigma^*, \tau^*) = \mathsf{true}$, it is necessary that also $\mathrm{Vf}(sk, m^*\|\sigma_2, \sigma_4) = \mathsf{true}$ holds. Therefore, in this case the only way that $(m^*\|\sigma_2, \sigma_4)$ is not a successful forgery of Σ is that $\mathcal{B}$ has called its signing oracle with message $m^*\|\sigma_2$ and obtained σ_4 as the reply. The implies that $\mathcal{A}$ has made a signing query (m^*, τ), and $H(\tau) = \sigma_2$. But since $\mathrm{Vf}'(sk', m^*, \sigma^*, \tau^*) = \mathsf{true}$, also $\sigma_2 = H(\tau^*)$. From the assumption, it follows that $\tau = \tau^*$, and in that case the signing query $(m^*, \tau) = (m^*, \tau^*)$ of $\mathcal{A}$ must have answered as $((pk\|\sigma_4) \oplus G(\tau^*))\|H(\tau^*) = \sigma^*$, which means that (m^*, σ^*, τ^*) is not new, so invalid, but this is contradiction.

Therefore, whenever $\mathcal{A}$ successfully forges for Σ', as long as L does not contain τ with $\tau \neq \tau^*$ and $H(\tau) = H(\tau^*)$, $\mathcal{B}$ also forges successfully for Σ. But the probability that L contains such an element in the simulation of $\mathcal{B}$ is precisely the success probability of $\mathcal{C}$, because $\mathcal{C}$ provides the identical simulation for $\mathcal{A}$ as $\mathcal{B}$. This proves the claimed inequality.

Next, we show that Σ' satisfies anonymity with respect to full key exposure. Suppose that $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ is an adversary attacking anonymity of Σ'. Using $\mathcal{A}$, we construct $\mathcal{B}$ attacking pseudorandomness of G with respect to H, satisfying

$$\mathbf{Adv}^{\mathsf{anon\text{-}fke}}_{\Sigma',\mathcal{A}}(k) = 2\mathbf{Adv}^{\mathsf{prg}}_{G,H,\mathcal{B}}(k).$$

Consider experiment $\mathbf{Expr}^{\mathsf{prg}\text{-}b}_{G,H,\mathcal{B}}$ with respect to this adversary $\mathcal{B}$. A random $\tau \xleftarrow{\mathrm{R}} \{0,1\}^{l_0(k)}$ is chosen, R_0 and R_1 are defined as $R_0 \xleftarrow{\mathrm{R}} \{0,1\}^{l_p(k)+l_s(k)}$, $R_1 \leftarrow G(\tau)$, and $(1^k, \rho_1, \rho_2) = (1^k, R_b, H(\tau))$ is given to $\mathcal{B}$.

$\mathcal{B}$ flips a coin $b' \xleftarrow{\mathrm{R}} \{0,1\}$, and provides the following simulation for $\mathcal{A}$. Using the security parameter k, $\mathcal{B}$ generates the common parameter P, and two key pairs (pk'_0, sk'_0), (pk'_1, sk'_1), and runs $\mathcal{A}_1(pk'_0, pk'_1, sk'_0, sk'_1)$, which halts with output (m^*, st). Then $\mathcal{B}$ computes σ'^* as follows:

$$\sigma'^* \leftarrow ((pk_{b'}\|\mathrm{Sig}(sk_{b'}, m^*\|\rho_2)) \oplus \rho_1)\|\rho_2.$$

$\mathcal{B}$ runs $\mathcal{A}_2(\sigma'^*, st)$, which halts with output b''. If $b'' = b'$, then $\mathcal{B}$ outputs 1, and otherwise $\mathcal{B}$ outputs 0 and halts.

Note that when $b = 1$, $\mathcal{B}$ provides to $\mathcal{A}$ a perfect simulation of anonymity experiment with full key exposure, with respect to b'. On the other hand, when $b = 0$, ρ_1 is an independent uniform random bitstring so σ'^* does not give any information about the bit b'. Therefore, when $b = 1$, the probability that $\mathcal{B}$ outputs 1 is equal to

$$\frac{1}{2}\left(\mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{anon\text{-}fke\text{-}1}}_{\Sigma',\mathcal{A}}(k) = 1\right] + \mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{anon\text{-}fke\text{-}0}}_{\Sigma',\mathcal{A}}(k) = 0\right]\right).$$

And when $b = 0$, the output b'' of $\mathcal{A}$ should be independent from b', hence the probability that $\mathcal{B}$ outputs 1 is equal to

$$\begin{aligned}&\frac{1}{2}\left(\mathbf{Pr}[b'' = 1 \,|\, b' = 1] + \mathbf{Pr}[b'' = 0 \,|\, b' = 0]\right)\\ =&\frac{1}{2}\left(\mathbf{Pr}[b'' = 1] + \mathbf{Pr}[b'' = 0]\right) = \frac{1}{2}.\end{aligned}$$

Then,

$$\begin{aligned}\mathbf{Adv}^{\mathsf{prg}}_{G,H,\mathcal{B}}(k) &= \left|\mathbf{Pr}[\mathbf{Expr}^{\mathsf{prg\text{-}1}}_{G,H,\mathcal{B}}(k) = 1] - \mathbf{Pr}[\mathbf{Expr}^{\mathsf{prg\text{-}0}}_{G,H,\mathcal{B}}(k) = 1]\right|\\ &= \left|\frac{1}{2}\left(\mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{anon\text{-}fke\text{-}1}}_{\Sigma',\mathcal{A}}(k) = 1\right] + \mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{anon\text{-}fke\text{-}0}}_{\Sigma',\mathcal{A}}(k) = 0\right]\right) - \frac{1}{2}\right|\\ &= \frac{1}{2}\left|\mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{anon\text{-}fke\text{-}1}}_{\Sigma',\mathcal{A}}(k) = 1\right] - \mathbf{Pr}\left[\mathbf{Expr}^{\mathsf{anon\text{-}fke\text{-}0}}_{\Sigma',\mathcal{A}}(k) = 1\right]\right|\\ &= \frac{1}{2}\mathbf{Adv}^{\mathsf{anon\text{-}fke}}_{\Sigma',\mathcal{A}}(k).\end{aligned}$$

Finally, let us show that Σ' satisfies unpretendability with respect to full key exposure. Suppose that $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ is an adversary attacking unpretendability of Σ'. Using $\mathcal{A}$, we construct an adversary $\mathcal{B}$ attacking collision resistance of H. $\mathcal{B}$ is given the security parameter k, and using this, generates common parameters and two key pairs (pk', sk'), (pk'^*, sk'^*), and runs $\mathcal{A}_1(pk'^*, sk'^*, pk', sk')$ and obtains an output (m^*, st). $\mathcal{B}$ then randomly picks $\tau^* \xleftarrow{\mathrm{R}} \{0,1\}^{l_0(k)}$, computes $\sigma'^* \leftarrow \mathrm{Sig}'(sk'^*, m^*, \tau^*)$, and runs $\mathcal{A}_2(\sigma'^*, \tau^*, st)$ and obtains an output τ. Then $\mathcal{B}$ halts with output (τ, τ^*). This simulation of the full-key exposure unpretendability experiment for $\mathcal{A}$ by $\mathcal{B}$ is perfect.

In order to analyze the advantage of $\mathcal{B}$, let us define the following function:

$$\mathrm{PC}_{\Sigma}(k) \stackrel{\mathrm{def}}{=} \mathbf{Pr}[pk = pk^* \,|\, P \leftarrow \mathrm{Par}(1^k); (pk, sk) \leftarrow \mathrm{Gen}(P); (pk^*, sk^*) \leftarrow \mathrm{Gen}(P)].$$

$\mathrm{PC}_{\Sigma}(k)$ measures the probability of public-key collision, which has to be negligible in any sane signature scheme. Indeed, if $\mathrm{PC}_{\Sigma}(k)$ is not negligible, then an adversary may generate his own key pair (pk', sk'), and compute a signature $\sigma \leftarrow \mathrm{Sig}(sk', m)$ which has non-negligible probability of passing verification with respect to pk, when $pk = pk'$.

We show that

$$\mathbf{Adv}^{\mathsf{up\text{-}fke}}_{\Sigma',\mathcal{A}}(k) \leq \mathbf{Adv}^{\mathsf{cr}}_{H,\mathcal{B}}(k) + \mathrm{PC}_{\Sigma}(k).$$

We claim that, in the above simulation, whenever $\mathcal{A}$ succeeds breaking unpretendability of Σ', that is, $\mathrm{Vf}'(pk', m^*, \sigma'^*, \tau) = \mathsf{true}$, then either $pk = pk^*$, or $\mathcal{B}$ also succeeds breaking collision resistance of H. Indeed, from the definition of Σ', $\sigma'^* = (pk^* \| \mathrm{Sig}(sk^*, m^* \| H(\tau^*))) \oplus G(\tau^*) \| H(\tau^*)$ holds, and in order that $\mathrm{Vf}'(pk', m^*, \sigma'^*, \tau) = \mathsf{true}$, it is necessary that $H(\tau) = H(\tau^*)$. Now, if $\tau \neq \tau^*$, then the output (τ, τ^*) of $\mathcal{B}$ is a valid collision pair. On the other hand, if $\tau = \tau^*$, then in order that $\mathrm{Vf}'(pk', m^*, \sigma'^*, \tau) = \mathsf{true}$, it is necessary that $pk = pk^*$. □

5.3 Boneh-Boyen short signature

Here we give a brief description of the Boneh-Boyen signature scheme for completeness.

Parameter generation A bilinear group $(\mathbb{G}_1, \mathbb{G}_2)$ with a pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, where $|\mathbb{G}_1| = |\mathbb{G}_2| = |\mathbb{G}_T| = p$ for some prime p, is chosen. The message space is $\mathbb{Z}_p$, which gives no essential problem since the domain can be extended by using a target collision resistant hash function.

Key generation Key generation algorithm chooses random generators g_1 and g_2 of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and chooses x, $y \xleftarrow{\mathrm{R}} \mathbb{Z}_p^*$, computes $u \leftarrow g_2^x \in \mathbb{G}_2$, $v \leftarrow g_2^y \in \mathbb{G}_2$, $z \leftarrow e(g_1, g_2) \in \mathbb{G}_T$. Then, $pk \stackrel{\mathrm{def}}{=} (g_1, g_2, u, v, z)$, and $sk \stackrel{\mathrm{def}}{=} (g_1, x, y)$.

Signing For a secret key (g_1, x, y) and a message $m \in \mathbb{Z}_p$, the signing algorithm chooses $\tau \xleftarrow{\mathrm{R}} \mathbb{Z}_p \setminus \{-\frac{x+m}{y}\}$, and computes $\sigma \leftarrow g_1^{1/(x+m+y\tau)} \in \mathbb{G}_1$. Then the signature is the pair (σ, τ).

Verification For a public key (g_1, g_2, u, v, z), a message m, and a signature (σ, τ), the verification can be done by checking whether $e(\sigma, u \cdot g_2^m \cdot v^\tau) = z$.

5.4 Security of Boneh-Boyen as an anonymous signature

The Boneh-Boyen short signature can be naturally considered as an anonymous signature, by regarding τ in $(\sigma = g_1^{1/(x+m+y\tau)}, \tau)$ as the verification token. To be precise, because τ should not be equal to $-(x+m)/y$ modulo p, we need to make slight modifications both to the signature scheme and to the formalism itself; for example, instead of choosing τ uniformly from $\mathbb{Z}_p \setminus \{-(x+m)/y\}$, τ may be chosen uniformly from $\mathbb{Z}_p$, and instead the signing algorithm may be allowed to fail in the negligible possiblity that $\tau = -(x+m)/y$.

Then, the Boneh-Boyen short signature scheme becomes a secure anonymous signature scheme. In fact, it satisfies the strongest security properties; it is strongly unforgeable, anonymous with full key exposure, and unpretendable with full key exposure.

Strong unforgeability Because our definition of strong unforgeability for anonymous signatures is identical to the ordinary definition of strong unforgeability, the proof of Boneh and Boyen for the strong unforgeability of the short signature scheme is directly applicable. Their proof is based on the SDH assumption on bilinear groups $(\mathbb{G}_1, \mathbb{G}_2)$.

Anonymity with full key exposure For a message $m \in \mathbb{Z}_p$ chosen by the adversary, consider the distribution of the signature σ, where $\sigma = g_1^{1/(x+m+y\tau)}$, for uniformly chosen token $\tau \xleftarrow{R} \mathbb{Z}_p$, when the secret key (g_1, x, y) is given to the adversary. Then, even conditioned on g_1, x, m, and y, still $1/(x+m+y\tau)$ has uniform distribution on $\mathbb{Z}_p^* \cup \{\perp\}$, and σ has uniform distribution on $(\mathbb{G}_1 \setminus \{1\}) \cup \{\perp\}$. Because this is true for any secret key (g_1, x, y), we conclude that the Boneh-Boyen short signature scheme is anonymous with full key exposure.

Unpretendability with full key exposure We will prove unpretendability of Boneh-Boyen signature with full key exposure, under the discrete logarithm assumption on the group $\mathbb{G}_1$.

Suppose that $\mathcal{A}$ is an unpretendability adversary. Using $\mathcal{A}$, we may construct a discrete logarithm solver $\mathcal{D}$ which, given $(g, h = g^\alpha)$ for a generator g of $\mathbb{G}_1$ and $\alpha \xleftarrow{R} \mathbb{Z}_p^*$, finds the discrete logarithm $\alpha = \log_g(h)$ as follows.

The solver $\mathcal{D}$ gives simulation of the unpretendability experiment for the adversary $\mathcal{A}$. $\mathcal{D}$ chooses $\beta \xleftarrow{R} \mathbb{Z}_p^*$, defines $g_1 \leftarrow h^\beta$, $g_1^* \leftarrow g^\beta$, and chooses g_2, g_2^* as random generators of $\mathbb{G}_2$. $\mathcal{D}$ then chooses x, y, x^*, $y^* \xleftarrow{R} \mathbb{Z}_p^*$, defines $u \leftarrow g_2^x$, $u^* \leftarrow (g_2^*)^{x^*}$, $v \leftarrow g_2^y$, $v^* \leftarrow (g_2^*)^{y^*}$, $z \leftarrow e(g_1, g_2)$, and $z^* \leftarrow e(g_1^*, g_2^*)$. Then $pk \stackrel{\text{def}}{=} (g_1, g_2, u, v, z)$, $sk \stackrel{\text{def}}{=} (g_1, x, y)$, $pk^* \stackrel{\text{def}}{=} (g_1^*, g_2^*, u^*, v^*, z^*)$, $sk^* \stackrel{\text{def}}{=} (g_1^*, x^*, y^*)$, and (pk, sk) and (pk^*, sk^*) are given to $\mathcal{A}$. Note that these key pairs are chosen with the identical distribution as in the original experiment. Also, we give the secret keys sk, sk^* to the adversary because here we are considering the full key exposure.

The adversary $\mathcal{A}$ will output the target message $m^* \in \mathbb{Z}_p$. Then the solver $\mathcal{D}$ chooses $\tau^* \xleftarrow{R} \mathbb{Z}_p$, computes $\sigma^* \leftarrow (g_1^*)^{1/(x^*+m^*+y^*\tau^*)}$, and returns (σ^*, τ^*) to $\mathcal{A}$. After more computation, $\mathcal{A}$ will output τ. Using the output τ, $\mathcal{D}$ outputs the following:

$$\alpha' \leftarrow \frac{x + m^* + y\tau}{x^* + m^* + y^*\tau^*}.$$

Let us prove that $\alpha' = \alpha$ whenever the output τ of $\mathcal{A}$ is a successful unpretendability attack. Suppose that is the case. This means that (m^*, σ^*, τ) passes the verification with respect to the public key pk. This is equivalent to the condition that $(\sigma^*)^{x+m^*+y\tau} = g_1$. But also from the definition of σ^*, we have $(\sigma^*)^{x^*+m^*+y^*\tau^*} = g_1^*$. Since $g_1 = h^\beta$ and $g_1^* = g^\beta$ by the definition, it follows that

$$h = g^\alpha = g^{\frac{x+m^*+y\tau}{x^*+m^*+y^*\tau^*}}.$$

This proves that $\mathcal{D}$ outputs the correct discrete logarithm $\log_g(h)$ whenever $\mathcal{A}$ succeeds the unpretendability attack.

6 Conclusion

We re-examined the formal definition of an anonymous signature first proposed by Yang et al. [7], and showed that the formalism is inadequate in a few aspects. We fixed the formalism by relying on an explicitly given randomness instead of hidden residual randomness in the message, and moved it from the message to the signature itself. We also identified a crucial property of an anonymous signature which we call unpretendability. We realized our definition, by providing a generic construction out of any ordinary signature scheme. Finally, we examined the short signature of Boneh and Boyen [3], and showed that it can be naturally regarded as an anonymous signature, which is provably secure in the standard model.

References

1. Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In Colin Boyd, editor, *ASIACRYPT*, volume 2248, pages 566–582. Springer, 2001.
2. Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM Journal on Computing*, 13(4):850–864, 1984.
3. Dan Boneh and Xavier Boyen. Short signatures without random oracles and the SDH assumption in bilinear groups. *Journal of Cryptology*, 21(2):149–177, 2008.
4. Colin Boyd and Dong-Gook Park. Public key protocols for wireless communications. In *ICISC*, pages 47–57. Korea Institute of Information Security and Cryptology (KIISC), 1998.
5. Marc Fischlin. Anonymous signatures made easy. In Tatsuaki Okamoto and Xiaoyun Wang, editors, *Public Key Cryptography*, volume 4450, pages 31–42. Springer, 2007.
6. Steven D. Galbraith and Wenbo Mao. Invisibility and anonymity of undeniable and confirmer signatures. In Marc Joye, editor, *CT-RSA*, volume 2612, pages 80–97. Springer, 2003.
7. Guomin Yang, Duncan S. Wong, Xiaotie Deng, and Huaxiong Wang. Anonymous signature schemes. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin, editors, *Public Key Cryptography*, volume 3958, pages 347–363. Springer, 2006.
8. Rui Zhang and Hideki Imai. Strong anonymous signatures. In Moti Yung, Peng Liu, and Dongdai Lin, editors, *Inscrypt*, volume 5487, pages 60–71. Springer, 2008.