A Framework for Unique Ring Signatures

Matthew Franklin [*]   Haibin Zhang [†]

March 23, 2017

Abstract

We propose a simple, general, and unified framework for constructing unique ring signatures that simplify and capture the spirit of linkable ring signatures. The framework, which can be efficiently instantiated in the random oracle and the standard model, is obtained by generalizing the Bellare-Goldwasser "PRF made public" paradigm. Security of the first instantiation can be more tightly related to the CDH problem and the DDH problem, compared to prior linkable ring signatures. The scheme leads to the most efficient linkable ring signature in the random oracle model, for a given level of provable security. The second one based on stronger assumptions partly simplifies and slightly improves the sublinear size traceable ring signature of Fujisaki (CT-RSA 2011).

Keywords: anonymity, provable security, ring signature, tight reduction, unique signature, verifiable random functions.

1 Introduction

Ring signatures [44] are very useful tools for many privacy-preserving applications. However, they are not adequate in settings where some degree of privacy for users must be balanced against limited access. For example, a service provider might have the list of public keys that correspond to all users that have purchased a single access to some confidential service for that day (requiring anonymous authentication). For this kind of application, a number of restricted-use ring signatures are proposed. Notable examples include *linkable ring signatures* [1, 19, 37, 38, 48, 49] and *traceable ring signatures* [27, 28].

Linkable ring signature asks that if a user signs any two messages (same or different) with respect to the same ring, then an efficient public procedure can verify that the signer was the same (although the user's identity is not revealed).

Traceable ring signature is a ring signature scheme where each message is signed not only with respect to a list of ring members, but also with respect to an *issue* (e.g., identifying label of a specific election or survey). If a user signs any two different messages with respect to the same list of ring members *and* the same issue label, then the user's identity is revealed by an efficient public procedure. If a user signs the same message twice with respect to the same list of ring members *and* the same issue label, then the two signed messages can be determined to have come from the same signer by an efficient public procedure (although the signer's identity remains concealed).

[*] Department of Computer Science, University of California at Davis, Davis, California, 95616, USA. E-mail: franklin@cs.ucdavis.edu WWW: http://www.cs.ucdavis.edu/∼franklin/

[†] Department of Computer Science, University of California at Davis, Davis, California, 95616, USA. E-mail: hbzhang@cs.ucdavis.edu WWW: http://csiflabs.cs.ucdavis.edu/∼hbzhang/

Both linkable ring signatures and traceable ring signatures admit interesting applications such as various *e-voting systems* (e.g., *k-candidate, weighted sum, ranked choice, approval*, and *receipt-free voting* [18], etc.), *e-token systems* [13] (that generalizes *unclonable authentication* [22] and *k-times anonymous authentication* [41, 46, 47]), and so on. Notably, the e-voting schemes *directly* from linkable or traceable ring signatures do *not* need any central authorities, a unique and desirable property in sharp contrast to all the schemes from other methods.

Unique ring signatures. We define *unique ring signatures* that capture the essence of linkable ring signatures and traceable ring signatures without identity revelation. We may say a ring signature scheme *unique* if whenever a signer produces two different ring signatures of the *same message* with respect to the same ring, such that both will pass the verification procedure, then these two ring signatures will always have a large common component (hereinafter *unique identifier*). For all the applications introduced in this paper, we further need a *non-colliding* property for a unique ring signature. Call a unique ring signature non-colliding if two different signers of the same message, almost never produce ring signatures with the same unique identifier.

Definitional contributions. Linkable ring signature in essence exploits meaningful linkability in the setting of ring signature. Intuitively, the security notions, following the refined formulation due to Bender, Katz, and Morselli [7], include unforgeability, restricted anonymity (due to the linking procedure), and secure linkability. The last notion, simply speaking, asks that the signatures by the same user should be linked. This is indeed the perspective that many early papers [37, 38, 48, 49] take. But this alone is not adequate, for an adversary, obtaining the secret key of some user, might be able to produce a signature that is linked with a given one. This issue, considered by follow-on work [1, 19] as well as highlighted in [28, Appendix D], can be a serious one.[1]

Our formulation simplifies the definitions of security for linkable ring signatures without losing the power and generality of the primitive. We take a *different* approach, following [26], to formalizing the overall security notions. In a nutshell, it is required that each signer only sign any message once. More precisely, a set of signers in a ring cannot produce signatures for any messages with more unique identifers than the size of the set.

It turns out, however, that besides anonymity and unforgeability, the only definition of security that we need for unique ring signature is *uniqueness*. Together with non-collision property (which is *not* a security notion), our primitive is as powerful as linkable ring signature.[2] It is also easily seen that insider attacks are avoided if the signatures are unique, since otherwise one can construct another adversary violating the uniqueness property.

Our technical contributions. We propose a simple, general, and unified construction for unique ring signature, mainly by extending the "PRF made public" paradigm by Bellare and Goldwasser (BG) [4]. The signature scheme simply uses a combination of pseudorandom function (PRF) and non-interactive zero-knowledge (NIZK) proof system (where the PRF key is committed). The general framework not only can help explain prior constructions for linkable ring signatures and traceable ring signatures, but give refined constructions with simpler and more intuitive design and improved efficiency. We comment that the simple framework is partly motivated by the formulation of the unique ring signature, since both of (certified) PRF and unique ring signature enjoy uniqueness and pseudorandomness (a notion closely related to"anonymity").

Given the general framework, we first provide an efficient instantiation in the random oracle

[1] The corresponding security notion is formally called *non-slanderability* in [1], while the attack is termed as *insider attacks* [28] to indicate that the adversary might obtain a valid signing key.

[2] Indeed, one can safely regard that unique ring signature is functionally the same as linkable ring signatures, but definitionally more concise and simple, and furthermore, as shown shortly, more suited for our constructions.

model (ROM). Compared to prior linkable ring signatures, security of the scheme can be more tightly reduced to the CDH problem and the DDH problem (where, by "tight," it means that the success probability of some adversary in some time is roughly equal to the probability of solving some hard problem within almost the same period of time). Despite the similarities with the linkable ring signature due to Liu, Wei, and Wong [37], our scheme also exploits the *algebraic* property of the CDH and DDH problems, namely, the random self-reducibility (RSR) property (see, e.g., [2]) and uses Coron's technique [20].

We go on to illustrate the usefulness and generality of our framework by showing how to obtain a unique ring signature scheme from the traceable ring signature due to Fujisaki [28]. The latter is the first traceable ring signature (and linkable ring signature) without random oracles, and has a signature of size $\mathcal{O}(\sqrt{n})$ where n is the number of users in the ring. Our scheme is not simply a weakened version of [28] that removes the extra public tracing functionality. Fujisaki's scheme is based on the ring signature due to Chandran, Groth, and Sahai [14], while our scheme follows *exactly* our general framework, simplifying and clarifying the overall structure, eliminating the relatively inefficient one-time signature, employing a solo assumption (i.e., Pseudo-Random DDHI assumption [28]), and requiring *no* proofs any more (as impled by the general framework).

Our work improves the state of the art in unique/linkable ring signatures, thus leading to numerous improved e-voting and e-token systems from them.

Provable secure signatures and Tight reduction—History, Philosophy, and Approaches. Typically, one evaluates provably secure signature schemes from *three* perspectives: *efficiency*, indicating how fast the scheme can be implemented, which has an immediate impact on its genuine utility; *concrete security reduction*, which gives explicit bounds on success probability of the adversary, enabling meaningful comparisons for a given level of provable security; and *cryptographic assumptions*, preferably being simple, standard, and well-studied, on which the security of the scheme relies. A *desirable* provably secure cryptographic signature, commonly recognized, whether in the random oracle standard or the standard model, should be *at first* efficient, and could be *as well* tightly related to a reasonable assumption. Of course, it is also desirable to consider various tradeoffs among the three factors, provided that the scheme is still sufficiently efficient.

For signature schemes based on discrete logarithm problems, the most efficient scheme is the Schnorr signature [45] that is proven secure in the ROM under the DL assumption by Pointcheval and Stern [43]. The technique used is the Forking Lemma: by *rewinding* the forger $\mathcal{O}(q_h/\varepsilon)$ times, where q_h denotes the number of the forger makes to the random oracles and ε denotes its success probability one can compute the discrete logarithm of the public key. The reduction is unfortunately too loose. To obtain tight security reductions for the DL-based signature schemes, a number of constructions that are less efficient or/and under *stronger* assumptions are proposed, including the EDL scheme by Goh and Jarecki [29] (under the CDH assumption), subsequent work by Chevallier-Mames [17] (under the CDH assumption), two schemes by Katz and Wang [36] (from the CDH and DDH problems respectively), and Fischlin's scheme [25] (that relies on the DL assumption but is relatively inefficient).

Turning to the DL-type ring signature schemes, tight reductions are more challenging to achieve. This is due, first, to the fact that all the DL based ones, to the best of our knowledge, follow the CDS paradigm [21] whose security seems to inevitably rely on the (generalized) rewinding technique (see, e.g., [35]). This is further due to the fact that the ring signature runs in the multi-user setting such that the reduction might *naturally* lose a factor of n which denotes the number of users in the ring.

The linkable ring signature [37] from the DDH assumption inherit the CDS framework and its analysis for ordinary ring signatures. In particular, if we let ε be an upper bound on the probability that the DL problem can be solved, then the success probability of any adversaries attacking the unforgeability is roughly $nq_h\varepsilon$, but for anonymity one has to rely on the potentially stronger DDH assumption. Similar results hold for the traceable ring signature [27], where Fujisaki and Suzuki therefore consider using Fischlin's technique [25, Remark 5.7] to improve the reduction tightness at a notable cost.

Our random oracle based scheme has security that can be more tightly reduced to the CDH and DDH problems. The unique ring signature does not exactly follows the framework (i.e., the public key, strictly speaking, is not a commitment) and the corresponding proof is thus non-black-box. Our scheme also exploits the *algebraic* property of the CDH and DDH problems, namely, the random self-reducibility (RSR) property (see, e.g., [2]) and uses Coron's technique [20].

Discussion. The deterministic and unique property of our unique ring signature can admit fast processing of data. For example, a service provider carrying out a "first come, first kept" policy on a stream of ℓ requests would need only $\mathcal{O}(\ell \log \ell)$ operations (via appropriate tree structures), or $\mathcal{O}(\ell)$ expected operations (via hash tables). It is (conceptually) in contrast to having to perform $\Theta(\ell^2)$ instances of the linking procedure in the general case to process a stream of ℓ requests. This is a particularly useful property when there is a large number of users to be processed. (Note that this essentially shares some similarity with public-key deterministic encryption [3].) Also applications using our methods would greatly save space complexity. Once a signature is verified, it just needs to save the unique identifier, which is one group element for all of constructions. Note that saving just these single group elements (or even just their hashes) is sufficient for carrying out the desired functionality.

In fact, there is one natural alternative, which we call *all-ring unique ring signature*, requiring the uniqueness to hold for all the rings (i.e., even with respect to different rings). The corresponding variant is implicit in the applications of linkable ring signatures (e.g., [49]), and is even considered as being weak or "flawed" due to its relaxed security. We are sympathetic to this viewpoint for most of the applications, but do point out some interesting observations. Basically, all-ring unique ring signature enables *flexible ring size choice* and *dynamic membership*, providing greater flexibility to *both* users and system providers, for a few certain applications such as the one in the last paragraph. On the one hand, it allows the signer to choose to hide within an arbitrary ring of authorized users. For instance, the user may not want to include some other specific users with bad reputation; it is also entirely possible that the user cares about efficiency issues, since the computational overhead and even the size of signature are proportional to the number of users in the ring. Therefore, the signer can choose an appropriate ring size to balance identity privacy concerns with computational overhead at her will. On the other hand, all-ring unique ring signature admits dynamic membership. In this setting, the public keys correspond to the membership of the users. The users have to pay to be maintained in the list of the service provider for some period. Once the time is up for some user, the provider can simply remove its public key from the list. The user can choose any subsets of the current list of public keys to form a ring and sign on some message, while the provider only accepts signatures with respect to rings that are subsets of the current list.

This version. Goldberg and Reyzin (March 2017) discovered that if one does not hash the unique identifier when computing the challenge of the proof system, the uniqueness of the VRF from DDH assumption is violated. Therefore, when using the VRF, it is important to hash the unique identifier as well.

Moreover, they also identified an important flaw on proving the unforgeability of our unique ring signature from DDH assumption. In the new version, we can only prove a more loose bound on the unforgeability for our unique ring signature in the ROM.

2 Preliminaries

Notations. If x is a string then $|x|$ denotes its length. If S is a set then $|S|$ denotes its size and $s \xleftarrow{\$} S$ denotes the operation of selecting an element s of S uniformly at random. $\emptyset$ denotes the empty set, while $\boldsymbol{\emptyset}$ denotes a vector of empty sets. If n is an integer we write $[n]$ to denote the set $\{1, 2, \cdots, n\}$. We let $\{B_i\}_{i=1}^{n}$ (or simply $\{B_i\}_1^n$) either denote the set $\{B_1, B_2, \cdots, B_n\}$ or $B_1||B_2||\cdots||B_n$ (the concatenation of $B_1, B_2, \cdots$, and B_n), where there should be no ambiguity from context. If $\mathcal{A}$ is a randomized algorithm then we write $z \xleftarrow{\$} \mathcal{A}(x, y, \cdots)$ to indicate the operation that runs $\mathcal{A}$ on inputs $x, y, \cdots$ and a uniformly selected r from an appropriately required domain and outputs z. We write $z \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_1, \mathcal{O}_2, \cdots}(x, y, \cdots)$ to indicate the operation that runs $\mathcal{A}$ having access to oracles $\mathcal{O}_1, \mathcal{O}_2, \cdots$ on inputs $x, y, \cdots$ and outputs z. A function $\epsilon(k)\colon \mathbb{N} \to \mathbb{R}$ is *negligible* if, for any positive number d, there exists some constant $k_0 \in \mathbb{N}$ such that $\epsilon(k) < (1/k)^d$ for any $k > k_0$.

2.1 Primitives

Pseudo-random function. We define a *pseudo-random function* [31] family $F\colon \mathcal{S} \times \mathcal{X} \to \mathcal{Y}$ where $\mathcal{S}$ is the *key space*, $\mathcal{X}$ is the *message space*, and $\mathcal{Y}$ is the *range*. We write $F_s(\cdot)$ to denote a PRF for every $s \in \mathcal{S}$. Let Γ be the set of all functions from $\mathcal{X}$ to $\mathcal{Y}$. Define the PRF advantage of $\mathcal{A}$ against F as

$$\mathbf{Adv}_F^{\mathrm{prf}}(\mathcal{A}) = \Pr[\, s \xleftarrow{\$} \mathcal{S} :\ \mathcal{A}^{F_s} = 1\,] - \Pr[\, f \xleftarrow{\$} \Gamma :\ \mathcal{A}^{f} = 1\,].$$

Digital signatures. A *digital signature* $\mathcal{DS}$ consists of three algorithms $(\mathsf{Gen}, \mathsf{Sig}, \mathsf{Vrf})$. A *key generation* algorithm Gen takes the security parameter λ and generates a *verification key* vk and a *signing key* sk. A *signing* algorithm Sig computes a signature σ for input message m using the signing key sk. A *verification* algorithm Vrf takes as input vk and a message-signature pair (m, σ) and outputs a single bit b. It is required that for all the messages m it holds that $\Pr[\mathsf{Vrf}(vk, m, \mathsf{Sig}(sk, m)) = 1] = 1$. The standard security notion of a digital signature is *existential unforgeability against adaptive chosen message attacks* [32]. Formally, given a signature scheme $\mathcal{DS}$, we associate to an adversary $\mathcal{A}$ the following experiment:

```
Experiment $\mathbf{Exp}_{\mathcal{DS}}^{\mathrm{uf}}(\mathcal{A})$
  $(vk, sk) \xleftarrow{\$} \mathcal{DS}.\mathsf{Gen}(1^\lambda)$
  $(m, \sigma) \xleftarrow{\$} \mathcal{A}^{\mathsf{Sig}(sk, \cdot)}(vk)$
  if $\mathsf{Vrf}(vk, m, \sigma) = 0$ then return $0$
  return $1$
```

where m was not a query of $\mathcal{A}$. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}_{\mathcal{DS}}^{\mathrm{uf}}(\mathcal{A}) = \Pr[\mathbf{Exp}_{\mathcal{DS}}^{\mathrm{uf}}(\mathcal{A}) = 1].$$

Verifiable random function. *Verifiable random function* (VRF), introduced by Micali, Rabin, and Vadhan [39], combines the properties of PRF and digital signature. Namely, a VRF is a PRF with a non-interactive *proof of the correctness* of the input. A VRF $\mathcal{VRF}$ consists of four algorithms $(\mathsf{Gen}, \mathsf{Eva}, \mathsf{Prove}, \mathsf{Ver})$ with *input domain* $\mathcal{X}$ and *output range* $\mathcal{Y}$. A *key generation* algorithm Gen takes the security parameter λ and outputs a pair of keys (vk, sk). An *evaluation* algorithm Eva

takes as input sk and some x and outputs a value y. A *proving* algorithm Prove takes as input sk and some x and outputs ν which is the *proof* of correctness. A *verification* algorithm Ver takes as input vk and (x, y, ν) and outputs a single bit b. Formally, we require:

- Provability/Correctness. If $y \leftarrow \mathsf{Eva}(sk, x)$ and $\nu \xleftarrow{\$} \mathsf{Prove}(sk, x)$ then $\mathsf{Ver}(vk, x, y, \nu) = 1$.
- Unconditional Uniqueness. There do not exist $(vk, x, y_1, y_2, \nu_1, \nu_2)$ such that $y_1 \neq y_2$, but $\mathsf{Ver}(vk, x, y_1, \nu_1) = \mathsf{Ver}(vk, x, y_2, \nu_2) = 1$. Note that uniqueness in the definition above can be relaxed so as to hold *computationally* as opposed to *unconditionally.*
- Pseudorandomness. We associate to an adversary $\mathcal{A}$ the following experiment:

```
$\mathbf{Experiment}\ \mathbf{Exp}^{\mathrm{pr}}_{\mathcal{VRF}}(\mathcal{A})$
  $(vk, sk) \xleftarrow{\$} \mathcal{VRF}.\mathsf{Gen}(1^\lambda)$
  $(x, \mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathsf{Eva}(sk,\cdot),\mathsf{Prove}(sk,\cdot)}(pk)$
  $y_0 \leftarrow \mathsf{Eva}(sk, x);\ y_1 \xleftarrow{\$} \mathcal{Y}$
  $b \xleftarrow{\$} \{0,1\};\ b' \xleftarrow{\$} \mathcal{A}(y_b, \mathsf{s})$
  $\mathbf{if}\ b' \neq b\ \mathbf{then\ return}\ 0$
  $\mathbf{return}\ 1$
```

  where the adversary did *not* query its oracles with x. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\mathrm{pr}}_{\mathcal{VRF}}(\mathcal{A}) = \Pr[\mathbf{Exp}^{\mathrm{pr}}_{\mathcal{VRF}}(\mathcal{A}) = 1] - 1/2.$$

  A VRF scheme $\mathcal{VRF}$ is said to have the pseudorandomness property if for any polynomial-time adversary $\mathcal{A}$ the function $\mathbf{Adv}^{\mathrm{pr}}_{\mathcal{VRF}}(\mathcal{A})$ is negligible in the security parameter. For our purposes, we need a stronger form of VRF such that the proof is zero-knowledge, i.e., PRF with a NIZK proof.

Commitment scheme. A *commitment* scheme $\mathcal{CM}$ consists of a randomized *committing* algorithm Com which takes as input a message m and randomness r to return a commitment c; we write $c \xleftarrow{\$} \mathsf{Com}(r, m)$. It is required that the commitment scheme have *hiding* and *binding* properties. We define the hiding-*advantage* of $\mathcal{A}$ against $\mathcal{CM}$ as

$$\begin{aligned}\mathbf{Adv}^{\mathrm{hide}}_{\mathcal{CM}}(\mathcal{A}) = \Pr[(m_0, m_1) \xleftarrow{\$} \mathcal{A}; c \xleftarrow{\$} \mathsf{Com}(r_1, m_1) :\ \mathcal{A}(c) = 1]\\ - \Pr[(m_0, m_1) \xleftarrow{\$} \mathcal{A}; c \xleftarrow{\$} \mathsf{Com}(r_0, m_0) :\ \mathcal{A}(c) = 1].\end{aligned}$$

We define the binding-*advantage* of $\mathcal{A}$ against $\mathcal{CM}$ as

$$\mathbf{Adv}^{\mathrm{bind}}_{\mathcal{CM}}(\mathcal{A}) = \Pr[(m_0, m_1, r_0, r_1) \leftarrow \mathcal{A}(ck) :\ m_0 \neq m_1 \text{ and } \mathsf{Com}(r_0, m_0) = \mathsf{Com}(r_1, m_1)].$$

Non-interactive zero-knowledge proof systems. We shall use a notion of *NIZK proof of membership* in *NP* languages, introduced by Blum, Feldman, and Micali [8]. Let $\rho(\cdot, \cdot)$ be a polynomially bounded binary relation. If $(x, w) \in \rho$ then x is a *theorem* and w is a *proof* of x. Let $\mathcal{L}_\rho$ denote the language associated with the relation ρ: $\mathcal{L}_\rho = \{x | \exists w[(x, w) \in \rho]\}$. Consider two polynomial-time algorithms (P, V), both of which have access to a *common reference string* η. (If the string is distributed uniformly at random then we will call it *common random string.*) Call (P, V) is a *non-interactive proof system* for $\mathcal{L}_\rho$ if there exists some polynomial $l(\cdot)$ such that it satisfies the following two conditions:

- **Completeness**: For every $\lambda \in \mathbb{N}$, every $(x, w) \in \rho$,
$$\Pr[\eta \overset{\$}{\leftarrow} \{0,1\}^{l(\lambda)}; \pi \overset{\$}{\leftarrow} P(\lambda, x, w, \eta): \ V(\lambda, x, \pi, \eta) = 1] = 1.$$
- **(Adaptive) soundness**: For every $\lambda \in \mathbb{N}$, any prover $\hat{P}$, and every $x \notin \mathcal{L}_\rho$,
$$\Pr[\eta \overset{\$}{\leftarrow} \{0,1\}^{l(\lambda)}; (x, \pi) \overset{\$}{\leftarrow} \hat{P}(\lambda, \eta): \ V(\lambda, x, \pi, \eta) = 1] \leq \epsilon(\lambda).$$

We let $\mathbf{Adv}^{\text{sound}}_{(P,V)}(\hat{P})$ denote the above soundness advantage of $\hat{P}$ against a non-interactive proof system (P, V).

Given a polynomial time simulator $S = (S_1, S_2)$, define the zero-knowledge advantage of $\mathcal{A}$ against a non-interactive proof system (P, V) as $\mathbf{Adv}^{\text{zk}}_{(P,V)}(\mathcal{A}) = \Pr[\eta \overset{\$}{\leftarrow} \{0,1\}^{l(\lambda)}; (x, w) \overset{\$}{\leftarrow} \mathcal{A}(1^\lambda, \eta); \pi \overset{\$}{\leftarrow} P(\lambda, x, w, \eta): \ \mathcal{A}(\lambda, x, \pi, \eta) = 1] - \Pr[(\eta', \mathsf{s}) \overset{\$}{\leftarrow} S_1(1^\lambda); (x, w) \overset{\$}{\leftarrow} \mathcal{A}(1^\lambda, \eta'); \pi' \overset{\$}{\leftarrow} S_2(x, \eta', \mathsf{s}) : \mathcal{A}(\lambda, x, \pi', \eta') = 1]$, where s is the state information. We say a non-interactive proof system (P, V) for $\mathcal{L}_\rho$ is (adaptive) zero-knowledge if there exists a probabilistic polynomial time simulator (S_1, S_2) such that for any probabilistic polynomial time adversary $\mathcal{A}$, it holds that $\mathbf{Adv}^{\text{zk}}_{(P,V)}(\mathcal{A}) \leq \epsilon(\lambda)$.

Non-interactive witness-indistinguishable proof systems. We also use *non-interactive witness-indistinguishable (NIWI) proof system.* We define the WI-*advantage* of $\mathcal{A}$ against a non-interactive proof system (P, V) for a language $\mathcal{L}_\rho$ as $\mathbf{Adv}^{\text{wi}}_{(P,V)}(\mathcal{A}) = \Pr[\eta \overset{\$}{\leftarrow} \{0,1\}^{l(\lambda)}; (x, w_0, w_1) \overset{\$}{\leftarrow} \mathcal{A}(1^\lambda, \eta); \pi \overset{\$}{\leftarrow} P(\lambda, x, w_0, \eta): \ \mathcal{A}(\lambda, x, \pi, \eta) = 1] - \Pr[\eta \overset{\$}{\leftarrow} \{0,1\}^{l(\lambda)}; (x, w_0, w_1) \overset{\$}{\leftarrow} \mathcal{A}(1^\lambda, \eta); \pi \overset{\$}{\leftarrow} P(\lambda, x, w_1, \eta): \ \mathcal{A}(\lambda, x, \pi, \eta) = 1]$, where we require that $(x, w_0), (x, w_1) \in \rho$. We say a non-interactive proof system (P, V) *witness indistinguishable*, if for any probabilistic polynomial time adversaries $\mathcal{A}$ it holds that $\mathbf{Adv}^{\text{wi}}_{(P,V)}(\mathcal{A}) \leq \epsilon(\lambda)$.

2.2 Complexity Assumptions

DDH Assumption. Consider a cyclic group $\mathbb{G}$ of prime order q with a generator g. Define the DDH-*advantage* of $\mathcal{A}$ against $\mathbb{G}$ as : $\mathbf{Adv}^{\text{ddh}}_{\mathbb{G}}(\mathcal{A}) = \Pr[x, y \overset{\$}{\leftarrow} \mathbb{Z}_q : \ \mathcal{A}(g, g^x, g^y, g^{xy}) = 1] - \Pr[x, y, z \overset{\$}{\leftarrow} \mathbb{Z}_q : \ \mathcal{A}(g, g^x, g^y, g^z) = 1]$. The DDH assumption states that for any probabilistic polynomial time adversary $\mathcal{A}$ its DDH-*advantage* is negligible in the security parameter.

BGN bilinear groups. We make use of bilinear groups of composite order introduced by Boneh, Goh, and Nissim [10] $(N, \mathbb{G}, \mathbb{G}_T, e, g)$ where $\mathbb{G}$ is a (multiplicative) cyclic group of composite order N ($N = pq$, and p and q are primes), and $\mathbb{G}_p$,$\mathbb{G}_q$ are its cyclic subgroup of order p, and subgroup of order q, respectively, and g, g_p, g_q are generators of $\mathbb{G}$, $\mathbb{G}_p$ and $\mathbb{G}_q$, respectively, and e: $\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is an efficiently computable bilinear map. In what follows we shall call this mathematical system a BGN bilinear group.

Subgroup Decision Assumption. Given a BGN bilinear group as described above, we say that the subgroup decision assumption holds if random elements from $\mathbb{G}$ and $\mathbb{G}_p$ are computationally indistinguishable. We define subgroup decision-*advantage* of $\mathcal{A}$ against BGN system as $\mathbf{Adv}^{\text{sda}}_{\text{BGN}}(\mathcal{A}) = \Pr[r \overset{\$}{\leftarrow} \mathbb{Z}^*_N, h \leftarrow g^r : \mathcal{A}(N, \mathbb{G}, \mathbb{G}_T, e, g, h) = 1] - \Pr[r \overset{\$}{\leftarrow} \mathbb{Z}^*_p, h \leftarrow g^{qr} : \mathcal{A}(N, \mathbb{G}, \mathbb{G}_T, e, g, h) = 1]$.

Pseudo-Random DDHI assumption in $\mathbb{G}_p$. We now recall the pseudo-random DDHI (PR-DDHI) assumption first formalized by Fujisaki [28]. Given a BGN bilinear group, we define that PR-DDHI *advantage* against $\mathcal{A}$ as $\mathbf{Adv}^{\text{pr-ddhi}}_{\mathbb{G}_p}(\mathcal{A}) = \Pr[x \overset{\$}{\leftarrow} \mathbb{Z}_p : \mathcal{A}^{\sigma(x,\cdot)}(p, \mathbb{G}_p) = 1] - \Pr[f \overset{\$}{\leftarrow} \Gamma : \mathcal{A}^f(p, \mathbb{G}_p)]$, where $\sigma(x, \cdot) = g_p^{1/(x+\cdot)}$ and Γ is the set of all functions from Z_p to $\mathbb{G}_p$. We say the PR-DDHI assumption holds in $\mathbb{G}_p$ if for all probabilistic polynomial time adversaries $\mathcal{A}$, the above advantage is negligible.

3 Unique Ring Signature Model

We begin by recalling the definition of a *ring signature* scheme $\mathcal{RS} = (\mathsf{RK}, \mathsf{RS}, \mathsf{RV})$ that consists of three algorithms:

- $\mathsf{RK}(1^\lambda)$. The randomized *user key generation* algorithm takes as input the security parameter λ and outputs a public key pk and a secret key sk.
- $\mathsf{RS}(sk, R, m)$. The probabilistic *ring signing* algorithm takes as input a user secret key sk, a ring R that is a set of public keys (such that $pk \in R$), and a message m to return a signature σ on m with respect to the ring R.
- $\mathsf{RV}(R, m, \sigma)$. The deterministic *ring verification* algorithm takes as input a ring R, a message m, and a signature σ for m to return a single bit b.

The following correctness condition is required: for any security parameter λ, any integer n, any $\{(pk_i, sk_i)\}_1^n \xleftarrow{\$} \mathsf{RK}(1^\lambda)$ (where now $R = \{pk_i\}_1^n$), any $i \in [n]$, and any m, it holds that $\mathsf{RV}(R, m, \mathsf{RS}(R, sk_i, m)) = 1$.

We consider *unique ring signature* where the signature should have the form of $(R, m, \sigma) = (R, m, \tau, \pi)$ where τ is the *unique identifier* for some message m and some signer i, and π is the rest of the signature. For our constructions, one may simply consider that τ is *the* signature, and π is the corresponding (maybe probabilistic) proof of correctness. Following the recent formulation for ring signature due to Bender, Katz, and Morselli [7], we define for unique ring signature three security requirements: uniqueness, anonymity, and unforgeability. The way we define uniqueness property largely follows from that for unique group signature [26], where the uniqueness security is coupled to a non-colliding property.

Notations. Fixing a ring $\{(pk_i, sk_i)\}_1^n$ with $T = \{pk_i\}_{i=1}^n$, we describe two oracles for defining the security notions: *user secret keys oracle* $\mathsf{USK}(\cdot)$, which an adversary can call to get the signing key sk_i of some user $i \in [n]$; *ring signing oracle* $\mathsf{RS}(\cdot, \cdot, \cdot)$, which an adversary can call to get a ring signature for honest user i with respect to some ring R and some message m, where $i \in [n]$, such that $pk_i \in R$, and the other public keys in R need not be in $T = \{pk_i\}_{i=1}^n$.

Let $\mathtt{CU}$ denote a set of corrupted users whose secret signing keys are given to the adversary. Let $\mathtt{RS}$ denote a set of ring, message, and signature triples queried via the $\mathsf{RS}(\cdot, \cdot, \cdot)$ oracle. We write $\mathtt{RS}_{R,m}$ to denote a set of users with which adversary calls $\mathsf{RS}(\cdot, R, m)$. We write $\mathtt{RS}_{\mathbf{R},\mathbf{M}}$, where $\mathbf{R}$ is a set of the rings and $\mathbf{M}$ is a set of messages queried, to denote a vector of sets with $\mathtt{RS}_{R,m}$ for each $R \in \mathbf{R}$ and $m \in \mathbf{M}$.

Uniqueness. In the setting of ring signatures, uniqueness property intuitively means that a set of colluding signers in a ring cannot produce signatures for any messages with more unique identifers than the size of the set. The adversary is thus given the *user secret keys oracle* $\mathsf{USK}(\cdot)$ for an arbitrary set of users, and *ring signing oracle* $\mathsf{RS}(\cdot, \cdot, \cdot)$. Given a unique ring signature $\mathcal{RS} = (\mathsf{RK}, \mathsf{RS}, \mathsf{RV})$, we associate to an adversary $\mathcal{A}$ the following experiment:

```
Experiment $\mathbf{Exp}^{\mathrm{unique}}_{\mathcal{RS},n}(\mathcal{A})$
    $\{(pk_i, sk_i)\}_1^n \xleftarrow{\$} \mathsf{RK}(1^\lambda)$; $\mathtt{CU} \leftarrow \emptyset$; $\mathtt{RS}_{\mathbf{R},\mathbf{M}} \leftarrow \boldsymbol{\emptyset}$ where $T \leftarrow \{pk_i\}_1^n$
    $(m, \sigma_1, \cdots, \sigma_{|\mathtt{CU} \cup \mathtt{RS}_{T,m}|+1}) \xleftarrow{\$} \mathcal{A}^{\mathsf{USK}(\cdot), \mathsf{RS}(\cdot,\cdot,\cdot)}(T)$
    for $i \leftarrow 1$ to $|\mathtt{CU} \cup \mathtt{RS}_{T,m}| + 1$ do
        if $\mathsf{RV}(T, m, \sigma_i) = 0$ then return $0$
    for $i, j \leftarrow 1$ to $|\mathtt{CU} \cup \mathtt{RS}_{T,m}| + 1$ do
```

```
        if $i \neq j$ and $\tau_i = \tau_j$ then return $0$
    return $1$
```

where, above, each σ_i is of the form (τ_i, π_i). We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\text{unique}}_{\mathcal{RS},n}(\mathcal{A}) = \Pr[\mathbf{Exp}^{\text{unique}}_{\mathcal{RS},n}(\mathcal{A}) = 1].$$

In the above experiment, adversary is expected to output *exactly* $|\mathtt{CU} \cup \mathtt{RS}_{T,m}| + 1$ *valid* signatures which have *distinct* unique identifiers with respect to the *same* message.

Notice that one could define the uniqueness security property just like that for unique group signature [26]: namely, adversary would be expected to output *new* valid signatures. However, our formulation here turns out to be a *weaker* one, in the sense that adversary is allowed to simply output the signatures from the ring signing oracle. While one has to adopt a more complex uniqueness notion in the setting of group signature (see [26] for detailed discussion), the slightly weaker uniqueness definition for ring signature is sufficient for all of the applications. Therefore, we shall use this weak but less restricted uniqueness notion throughout the paper.

NON-COLLIDING PROPERTY. Following a similar argument as [26], the above uniqueness notion alone is problematic *per se*. For example, it is possible that k signers ought to create $k-1$ unique identifiers for some messages as two of them collide, but a collusion of k signers might be able to output k unique identifiers. Clearly, this does not contradict our uniqueness security, but makes them sign messages beyond their own.

We say that a ring signature is *non-colliding* if any of two different (honest) signers (who follow the scheme specification) almost never produce the same *unique identifier* of the same message with respect to the same ring. One should think of this as a correctness property rather than a security notion. Formally, for all security parameter λ and integer n, all $\{(pk_i, sk_i)\}_1^n \xleftarrow{\$} \mathsf{RK}(1^\lambda)$ with $T = \{pk_i\}_1^n$, all $i, j \in [n]$ and $i \neq j$, and all message $m \in \{0,1\}^*$, it holds that

$$\Pr[(\tau_i, \pi_i) \xleftarrow{\$} \mathsf{RS}(sk_i, T, m); (\tau_j, \psi_j) \xleftarrow{\$} \mathsf{RS}(sk_j, T, m) :\ \tau_i = \tau_j] \leq \epsilon(\lambda).$$

Above, the probability is taken over the coins of the group key generation algorithm and group signing algorithm.

The uniqueness security notion together with non-colliding property captures the essence of uniqueness in the multi-user ring signature setting. First, it resolves the problem above: if the above-mentioned circumstance happens then an adversary who corrupted a set of group members can always *honestly* generate signatures *again* and pick "enough" signatures with different unique identifiers to attack the uniqueness property. Second, it is easy to verify that uniqueness implies any linking notions in the literature.

Anonymity. With the restraints of being unique, one cannot achieve the strongest anonymity notion of Bender, Katz, and Morselli [7]. This is clearly because of the inherent limitations of our (partly) deterministic signing process. However, we can target for the following notion of anonymity that is still quite strong. Formally, given a unique ring signature scheme $\mathcal{RS} = (\mathsf{RK}, \mathsf{RS}, \mathsf{RV})$, we associate to an adversary $\mathcal{A}$ the following experiment:

```
Experiment $\mathbf{Exp}^{\text{anon}}_{\mathcal{RS},n}(\mathcal{A})$
    $\{(pk_i, sk_i)\}_1^n \xleftarrow{\$} \mathsf{RK}(1^\lambda)$; $\mathtt{CU} \leftarrow \emptyset$; $\mathtt{RS}_{\mathbf{R},\mathbf{M}} \leftarrow \mathbf{\emptyset}$ where $T \leftarrow \{pk_i\}_1^n$
    $(i_0, i_1, R, m) \xleftarrow{\$} \mathcal{A}^{\mathsf{USK}(\cdot),\mathsf{RS}(\cdot,\cdot,\cdot)}(T)$
    $b \xleftarrow{\$} \{0,1\}$; $\sigma \xleftarrow{\$} \mathsf{RS}(sk_{i_b}, R, m)$
```

```
$b' \xleftarrow{\$} \mathcal{A}^{\mathsf{USK}(\cdot),\mathsf{RS}(\cdot,\cdot)}(\mathsf{guess}, \sigma, \mathsf{s})$
$\textbf{if } b' \neq b \textbf{ then return } 0$
$\textbf{return } 1$
```

where it is mandated that for each $d \in \{0,1\}$ we have $i_d \notin \mathtt{CU}$ and $i_d \notin \mathtt{RS}_{R,m}$. It may be required that $R \subseteq T$, but this is optional. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\mathrm{anon}}_{\mathcal{RS},n}(\mathcal{A}) = \Pr[\mathbf{Exp}^{\mathrm{anon}}_{\mathcal{RS},n}(\mathcal{A}) = 1] - 1/2.$$

The formulation provides the strongest possible anonymity definition that we can imagine in the context of unique ring signature.

Unforgeability. We can achieve the strongest unforgeability notion due to Bender, Katz, and Morselli [7]. More concretely, given a unique ring signature scheme $\mathcal{RS} = (\mathsf{RK}, \mathsf{RS}, \mathsf{RV})$, we associate to an adversary $\mathcal{A}$ the following experiment:

```
$\textbf{Experiment } \mathbf{Exp}^{\mathrm{uf}}_{\mathcal{RS},n}(\mathcal{A})$
    $\{(pk_i, sk_i)\}_1^n \xleftarrow{\$} \mathsf{RK}(1^\lambda)$; $\mathtt{CU} \leftarrow \emptyset$; $\mathtt{RS}_{\mathbf{R},\mathbf{M}} \leftarrow \boldsymbol{\emptyset}$ $\textbf{where } T \leftarrow \{pk_i\}_1^n$
    $(m, R, \sigma) \xleftarrow{\$} \mathcal{A}^{\mathsf{USK}(\cdot),\mathsf{RS}(\cdot,\cdot,\cdot)}(T)$
    $\textbf{if } \mathsf{RV}(R, m, \sigma) = 0 \textbf{ then return } 0$
    $\textbf{return } 1$
```

where it is required that $R \subseteq T \backslash \mathtt{CU}$ and $\mathcal{A}$ never queried $\mathsf{RS}(\cdot,\cdot,\cdot)$ with $(\cdot, R, m)$. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\mathrm{uf}}_{\mathcal{RS},n}(\mathcal{A}) = \Pr[\mathbf{Exp}^{\mathrm{uf}}_{\mathcal{RS},n}(\mathcal{A}) = 1].$$

All-ring unique ring signature model. We sketch the security definitions of all-ring unique ring signature schemes, which no longer asks the uniqueness property to only hold for the prescribed rings. It is easy to see that the uniqueness notion and non-colliding property can be modified accordingly. In terms of other security notions, there are two differences from those for regular unique ring signature: first, in the anonymity experiment, it is required now that $\mathcal{A}$ never query the $\mathsf{RS}(\cdot,\cdot,\cdot)$ oracle with $(i_0, \cdot, m)$ or $(i_1, \cdot, m)$; second, in the unforgeability experiment, the adversary is now not allowed to query the $\mathsf{RS}(\cdot,\cdot,\cdot)$ oracle with $(\cdot,\cdot,m)$. It is clear that the changes in both of the unforgeability and anonymity experiments actually impose the adversary more restrictions.

4 Unique Ring Signature from General Assumptions

In this section, we give a general construction of unique ring signature in the common random string model, mainly by extending the design paradigm of Bellare and Goldwasser (BG) [4].

Some intuition. The basic idea of the BG signature is to make PRF public using a publicly verifiable NIZK proof. Specifically, the authority pre-selects an encryption scheme $E_{pk}(\cdot)$ and a family of pseudorandom functions $F(\cdot)$. A signer publishes an encryption C of some randomly chosen message s using a randomness r (i.e., $C = E_{pk}(r, s)$). Now, the signer produces a signature on m as (m, τ, π) where $\tau \leftarrow F_s(m)$ and π is a NIZK proof such that $(pk, C, m, \tau) \in \mathcal{L}$ where the language $\mathcal{L} := \{(pk, C, m, \tau) | \exists (s, r)[C = E_{pk}(r, s) \text{ and } \tau = F_s(m)]\}$. If the underlying NIZK proof system (P, V) is adaptively zero-knowledge then the above scheme is unforgeable against chosen-message attacks. Note that the signature identifier τ on a message m is not necessarily unique as the signer may find another pair (r', s') such that $(r, s) \neq (r', s')$ while $E_{pk}(r, s) = E_{pk}(r', s')$. This problem can be easily solved by replacing the encryption scheme with a commitment scheme. We extend this scheme to construct a unique ring signature. The idea is a simple one. Every user now commits to its own public key. Given a pre-selected ring R, it simply produces a signature on message m as (R, m, τ, π), where $\tau \leftarrow F_s(m||R)$ is the unique identifier and π is a NIZK proof for an "or" language such that there exists one user who indeed uses its committed message as a key to apply the PRF to $m||R$. The construction is detailed in Figure 1.

$\mathsf{Setup}(1^\lambda)$.

The setup algorithm selects a common random string $\eta \xleftarrow{\$} \{0,1\}^{l(\lambda)}$, a PRF family $F : \mathcal{S} \times \mathcal{X} \to \mathcal{Y}$ where $\mathcal{S}$ is the key space, $\mathcal{X}$ is the message space, and $\mathcal{Y}$ is the range, and a string commitment scheme $\mathcal{CM}$ with a committing algorithm Com. It outputs the public parameters as

$$\mathsf{pp} = (\lambda, \eta, F, \mathcal{CM}).$$

$\mathsf{RG}(1^\lambda, \mathsf{pp})$.

The key generation algorithm for user i takes as input the parameter pp and computes a string commitment C_i on a randomly chosen s_i using randomness r_i (i.e., $C_i = \mathsf{Com}(r_i, s_i)$), and outputs the public key as $pk_i = (\mathsf{pp}, C_i)$ and the secret key as $sk_i = (\mathsf{pp}, s_i, r_i)$.

$\mathsf{RS}(sk_i, R, m)$.

To sign the message m in the ring $R = (pk_1, ..., pk_n)$, the signer i uses its secret key (s_i, r_i) to produce a signature as

$$(R, m, \sigma),$$

where $\sigma = (\tau, \pi)$ in which $\tau \leftarrow F_{s_i}(m||R)$ is the unique identifier and π is a publicly verifiable NIZK proof that $(\{C_j\}_{j=1}^n, R, m, \tau) \in \mathcal{L}_{\mathsf{OR}}$ where $\mathcal{L}_{\mathsf{OR}} := \{(\{C_j\}_{j=1}^n, R, m, \tau) | \exists (j, s_j, r_j)[C_j = \mathsf{Com}(r_j, s_j)$ and $\tau = F_{s_j}(m||R)]\}$.

$\mathsf{RV}(R, m, \sigma)$.

The verification algorithm first parses σ as (τ, π) and checks if π is a correct NIZK proof for the language $\mathcal{L}_{\mathsf{OR}}$.

Figure 1: **Unique ring signature from general assumptions in the common random string model**. In particular, $\mathcal{X}$ is the ring signature space, and $\mathcal{Y}$ is the unique identifier range.

Notice that the NIZK proof system must be zero-knowledge in adaptive, multi-prover, multi-theorem setting [4, 23]. We follow the terminology of [23] to call it adaptive NIZK. The following theorem establishes the security of the scheme in Figure 1:

Theorem 1 *If F is a PRF family, $\mathcal{CM}$ is a commitment scheme, and the underlying NIZK for NP-languages is adaptively zero-knowledge then the scheme described in Figure 1 is a secure unique ring signature scheme.* ▮

All-ring unique ring signature. The above unique ring signature can be rather easily converted to an all-ring unique ring signature. The input into the PRF should now not contain the ring information R. That is, user i with secret key (r_i, s_i) gets the ring signature as (R, m, τ, π) where $\tau = F_{s_i}(m)$ and π is an adaptive NIZK proof that $(\{C_j\}_{j=1}^n, m, \tau) \in \mathcal{L}_{\mathsf{OR}}$ where $\mathcal{L}_{\mathsf{OR}} = \{(\{C_j\}_{j=1}^n, m, \tau) | \exists (j, s_j, r_j)[C_j = \mathsf{Com}(r_j, s_j)$ and $\tau = F_{s_j}(m)]\}$. The verification algorithm can be modified accordingly. Looking ahead, the following two unique ring signature constructions that we shall describe shortly can be likewise modified to be all-ring unique ring signature schemes.

5 Unique Ring Signature in Random Oracle Model

We start by describing our basic underlying signature/VRF scheme, and then give the construction of unique ring signature. Notice that our proof techniques do not require *proof of knowledge* but heavily rely on zero-knowledge proof of *membership*. This is one of the main reasons our signature enjoys tight security reductions and thereby admits an improvement in efficiency for a given level of provable security.

5.1 The Underlying VRF Scheme.

The signature we shall describe is first predicated on a (well-known) observation that given a random public group element $y = g^x$, the function $F(m) := H(m)^x$ is a PRF, if we model the hash function $H(\cdot)$ as a random oracle.

Our scheme is furthermore based on a well-known zero-knowledge proof system for equality of discrete logarithm due to Chaum and Pederson [16]:

A prover and a verifier both know (g, h, y_1, y_2) with $g, h \neq 1$ and $y_1 = g^x$ and $y_2 = h^x$ for an exponent $x \in \mathbb{Z}_q$. A prover also knows the exponent x. They run the following protocol:

1. The prover chooses $r \xleftarrow{\$} \mathbb{Z}_q$ and sends $a \leftarrow g^r$, $b \leftarrow h^r$ to the verifier.
2. The verifier sends a challenge $c \xleftarrow{\$} \mathbb{Z}_q$ to the prover.[3]
3. The prover sends $t \leftarrow r - cx \bmod q$ to the verifier.
4. The verifier accepts iff $a = g^t y_1^c$ and $b = h^t y_2^c$.

The above protocol is a *sound* proof system but also *honest-verifier zero-knowledge* (HVZK). By using Fiat-Shamir transformation [27], it becomes a NIZK proof system if we model the hash function as a random oracle.

Given the above PRF and NIZK proof system, we apply the BG paradigm to obtain a VRF scheme depicted in Figure 2. (The scheme is in fact a PRF with a NIZK proof and of course a secure signature scheme.) Of course, the function that maps x to g^x is not a commitment scheme: the binding property is satisfied while the hiding property is not. This prevents us from following the general NIZK construction's proof strategy exactly.

5.2 Extending the VRF to Unique Ring Signature

Extending the underlying proof system. With the general framework for unique ring signature, the core protocol is to extend the underlying NIZK proof to an "or" language — a proof system that a unique identifier τ (for a message m and a ring R) has the same logarithm with respect to base $H(m||R)$ as one of the public keys $y_j := g^{x_j}$ ($j \in [n]$) with respect to base g. Assume, without loss of generality, $\log_{H(m||R)} \tau = \log_g y_i$ and the prover knows x_i. In particular, we use the following proof system between a prover and a verifier.

[3] More precisely, one can choose c from $\{0,1\}^k$ where $k < \lceil \log q \rceil$ is a security parameter related to the tightness of reduction.

$\mathsf{Setup}(1^\lambda)$.

The setup algorithm takes as input the security parameter λ and outputs a multiplicative group $\mathbb{G}$ of prime order q and a randomly chosen generator g of $\mathbb{G}$. It also provides two hash functions $H\colon \{0,1\}^* \to \mathbb{G}$ and $H'\colon \{0,1\}^* \to \mathbb{Z}_q$. It outputs the public parameters as

$$\mathsf{pp} = (\lambda, q, \mathbb{G}, H, H').$$

$\mathsf{Gen}(1^\lambda, \mathsf{pp})$.

The key generation algorithm takes as input the parameter pp and chooses $x \xleftarrow{\$} \mathbb{Z}_q$ and computes $y \leftarrow g^x$. It outputs the public key as $pk = y$ and the secret key as $sk = x$.

$\mathsf{Sig}(sk, m)$.

To sign the message m, the signer selects $r \xleftarrow{\$} \mathbb{Z}_q$ and computes

$$(m, H(m)^x, c, t),$$

where $c \leftarrow H'(g, m, y, H(m)^x, g^r, H(m)^r)$ and $t \leftarrow r - cx \bmod q$.

$\mathsf{Vrf}(sk, m, \sigma)$.

The verification algorithm first parses σ as (m, τ, c, t) and checks if

$$c = H'(g, m, y, \tau, g^t y^c, H(m)^t \tau^c).$$

Figure 2: **Efficient Signature/VRF from the DDH assumption in the random oracle model**. The algorithms are described in the context of digital signature. It is also a VRF scheme, where $\mathcal{VRF}.\mathsf{Eva}(sk, m) = H(m)^x$, $\mathcal{VRF}.\mathsf{Prove}(sk, m) = (c, t)$, and $\mathcal{VRF}.\mathsf{Ver}(m, \sigma) = \mathcal{DS}.\mathsf{Vrf}(m, \sigma)$.

1. For $j \in [n]$ and $j \neq i$, the prover selects $c_j, t_j \xleftarrow{\$} \mathbb{Z}_q$ and computes $a_j \leftarrow g^{t_j} y_j^{c_j}$ and $b_j \leftarrow H(m)^{t_j} (H(m)^{x_i})^{c_j}$; for $j = i$, the prover selects $r_i \xleftarrow{\$} \mathbb{Z}_q$ and computes $a_i \leftarrow g^{r_i}$ and $b_i \leftarrow H(m)^{r_i}$. It sends $\{a_j, b_j\}_1^n$ to the verifier.

2. The verifier sends a challenge $c \xleftarrow{\$} \mathbb{Z}_q$ to the prover.

3. The prover computes $c_i \leftarrow c - \sum_{j \neq i} c_j$ and $t \leftarrow r - c_i x_i \bmod q$, and sends $c_1, t_1, \cdots, c_n, t_n$ to the verifier.

4. The verifier accepts iff $a_j = g^{t_j} y_j^{c_j}$ and $b_j = H(m)^{t_j} \tau^{c_j}$ for every $j \in [n]$.

The above protocol combines the Chaum-Pederson (CP) technique for proving the equality of two discrete logarithms of [16] and Cramer-Damgård-Schoenmakers (CDS) transformation [21]. Since both of the conversions "preserve" the properties of Σ-protocols, the above system is a sound proof system,[4] and also an interactive honest-verifier zero-knowledge of membership. However, as far as we are concerned, its soundness property has never been used in any signature schemes related to the above proof system. (This is perhaps due to the fact no one needs this property in these schemes anyway.) We now prove that the above proof system is sound;[5] in particular, even an arbitrarily malicious prover P^* cannot convince the verifier to accept a false statement.

[4] Strictly speaking, Σ-protocols can be divided into two categories: Σ-protocols for proof of knowledge, and Σ-protocols for proof of membership. In particular, we can formally show, in the setting of proof of membership, the special soundness property implies that a Σ-protocol is always an interactive proof system.

[5] This is needed, since in a moment, we shall be providing the exact bound on the soundness property in the random oracle model.

Proof: The goal is to show that if $\log_{H(m)} \tau \neq \log_g y_j$ for every $j \in [n]$, then given any $\{a_j, b_j\}_1^n$ sent by P^* there is at most one value c for which P^* can respond correctly. Recall above that we let x_0 denote $\log_{H(m)} \tau$ and x_j denote $\log_g y_j$ for every $j \in [n]$. In this case, we have that $x_0 \neq x_j$ ($j \in [n]$). Given any $\{a_j, b_j\}_1^n$ (where we assume $a_j = g^{r_j}$ and $b_j = H(m)^{r'_j}$) sent to the verifier by a cheating prover, we have the following: if the verifier is to accept, then we must have that

$$c = \sum_1^n c_j, \tag{1}$$

and for every $j \in [n]$,

$$\begin{aligned} a_j &= g^{t_1} y_j^{c_j}, & (2) \\ b_j &= H(m)^{t_j} \tau^{c_j}. & (3) \end{aligned}$$

By (2) and (3) we obtain that for every $j \in [n]$,

$$\begin{aligned} r_j &= t_j + x_j c_j, & (4) \\ r'_j &= t_j + x_0 c_j. & (5) \end{aligned}$$

Noting that $x_0 \neq x_j$ for every $j \in [n]$, we have $c_j \leftarrow (r_j - r'_j)(x_o - x_j)^{-1} \bmod q$. According to equation (1), we can now conclude that there is at most one challenge which the cheating prover can respond to. Therefore, the verifier generates this challenge with probability $1/q$ and the proof for soundness now follows. ▮

If we turn the above system into a NIZK proof system by following Fiat-Shamir transformation through a hash function H' then one can check that the soundness property is bounded by q_h/q, where q_h denotes the number of times the adversary makes to the random oracle H'. Indeed, in this case, for any $\{a_j, b_j\}_1^n$ and any query $H(m, \{a_j, b_j\}_1^n)$ made by an adversary P^*, it follows from the above proof that there is at most one possible value of c satisfying the verification equations.

The unique ring signature in the ROM is described in Figure 3.

The following theorem establishes the security of the above scheme.

Theorem 2 *The scheme presented in this section is a unique ring signature in the random oracle model under the CDH and DDH assumptions.* ▮

We highlight the main results with respect to all the unique ring signature definitions of security. Let $\mathbf{Adv}_{\mathbb{G}}^{\text{ddh}}$ and $\mathbf{Adv}_{\mathbb{G}}^{\text{cdh}}$ be an upper bound on the probability that the DDH problem and the CDH problem in $\mathbb{G}$ can be solved respectively. Let q_s denote the number of times the adversary makes to the signing oracle. If we do not give the adversary the corruption oracle following [37], we have the following results about the security of our unique ring signature: $\mathbf{Adv}_{\mathcal{RS}}^{\text{anon}}(\mathcal{A}) \approx \mathbf{Adv}_{\mathbb{G}}^{\text{ddh}}(\mathcal{A}_2)$, $\mathbf{Adv}_{\mathcal{RS}}^{\text{uf}}(\mathcal{A}) \approx q_s \mathbf{Adv}_{\mathbb{G}}^{\text{cdh}}(\mathcal{A}_3)$, and $\mathbf{Adv}_{\mathcal{RS}}^{\text{unique}}(\mathcal{A}) \approx t q_s \mathbf{Adv}_{\mathbb{G}}^{\text{ddh}}(\mathcal{A}_1)$.[7] If we allow signer corruption oracle, all the scheme advantages for adversaries lose a factor of n, where n is the size of target ring.

[6] Note that for this unique ring signature, this is *not* a common reference string setup algorithm. The public parameters can all be determined by the security parameter. We use this notation only for consistency.

[7] Variable t denotes the number of signatures output by the adversary.

$\mathsf{Setup}(1^\lambda)$.[6]

The setup algorithm takes as input the security parameter λ and outputs a multiplicative group $\mathbb{G}$ of prime order q and a randomly chosen generator g of $\mathbb{G}$. It also provides two hash functions $H\colon \{0,1\}^* \to \mathbb{G}$ and $H'\colon \{0,1\}^* \to \mathbb{Z}_q$. It outputs the public parameters as

$$\mathsf{pp} = (\lambda, q, \mathbb{G}, H, H').$$

$\mathsf{RG}(1^\lambda, \mathsf{pp})$.

The key generation algorithm for user i takes as input the parameter pp and selects an element $x_i \xleftarrow{\$} \mathbb{Z}_q$ and computes $y_i \leftarrow g^{x_i}$. It outputs the public key as $pk_i = (\mathsf{pp}, y_i)$ and the secret key as $sk_i = (\mathsf{pp}, x_i)$.

$\mathsf{RS}(sk_i, R, m)$.

To sign the message m in the ring $R = (pk_1, ..., pk_n)$, the signer i with the secret key $sk_i = x_i$ generates the signature in the following way:

1. (Simulation step.) For $j \in [n]$ and $j \neq i$, select $c_j, t_j \xleftarrow{\$} \mathbb{Z}_q$ and compute $a_j \leftarrow g^{t_j} y_j^{c_j}$ and $b_j \leftarrow H(m||R)^{t_j} (H(m||R)^{x_i})^{c_j}$.
2. For $j = i$, select $r_i \xleftarrow{\$} \mathbb{Z}_q$ and compute $a_i \leftarrow g^{r_i}$ and $b_i \leftarrow H(m||R)^{r_i}$.
3. Let $c_i \leftarrow H'(m, R, H(m||R)^{x_i}, \{a_j, b_j\}_1^n) - \sum_{j \neq i} c_j \bmod q$ and $t_i \leftarrow r_i - c_i x_i \bmod q$.
4. Return $(R, m, H(m||R)^{x_i}, c_1, t_1, \cdots, c_n, t_n)$.

$\mathsf{RV}(R, m, \sigma)$.

On receiving the signature (R, m, σ), the verification algorithm first parses σ as $(\tau, c_1, t_1, \cdots, c_n, t_n)$ and checks if $\tau \in \mathbb{G}$ and

$$\textstyle\sum_1^n c_j = H'(m, R, \tau, \{g^{t_j} y_j^{c_j}, H(m||R)^{t_j} \tau^{c_j}\}_1^n).$$

Figure 3: **Unique ring signature from the CDH and DDH assumptions in the random oracle model**.

6 Unique Ring Signature Without Random Oracles

We now show how to obtain a unique ring signature scheme from the traceable ring signature of Fujisaki [28]. Fujisaki's scheme is based on the ring signature due to Chandran, Groth, and Sahai [14], while our scheme follows *exactly* our general framework, simplifying and clarifying the overall structure, eliminating the relatively inefficient one-time signature, employing a solo assumption (i.e., Pseudo-Random DDHI assumption [28]), and requiring *no* proofs any more (as impled by the general framework).

6.1 The Underlying VRF Scheme.

We begin with the description of a verifiable random function (actually a PRF with a NIZK proof) mainly by modifying the traceable ring signature in [28], based on which we propose a unique ring signature scheme. Before we proceed, we recall several building blocks from [9–11, 14, 28, 33, 34].

Boneh-Boyen signature. Boneh and Boyen [9] gave a weakly unforgeable signature scheme based on the strong Diffie-Hellman assumption in $\mathbb{G}_p$ [9]. Given a bilinear group of prime order $(p, \mathbb{G}_p, \mathbb{G}_{T_p}, e, g_p)$, the signer chooses $x \xleftarrow{\$} \mathbb{Z}_p^*$ as the secret key and takes $y \leftarrow g_p^x$ as the public key. To sign on $m \in \mathbb{Z}_p$, the signer produces $g_p^{1/(x+m)}$. Given some signature (m, σ), one can verify it by checking if $e(\sigma, y g_p^m) = e(g_p, g_p)$. It is easy to give a variant of the Boneh-Boyen signature in BGN bilinear groups of composite order under the strong DDH assumption in those groups. This is an essential assumption for the construction of [28], but is not needed for ours.

NIWI proof for $\mathcal{L}_{BB}$ [34]. Given a BGN bilinear group $(N, p, q, \mathbb{G}, \mathbb{G}_T, e, g)$, define $x_p := x^q$ for every $x \in \mathbb{G}$. We now define the following language: $\mathcal{L}_{BB} := \{(V, m, \delta) \in \mathbb{G} \times Z_n^* \times \mathbb{G} | \exists x \in Z_n[V_p = g_p^x \text{ and } \delta_p = g_p^{1/(x+m)}]\}$. According to [34], the NIWI proof for $(V, m, \delta) \in \mathcal{L}_{BB}$ is given as follows: given a common reference string $(N, \mathbb{G}, \mathbb{G}_T, e, g, h)$ where h is either randomly chosen from $\mathbb{G}_p$ or $\mathbb{G}$. The witnesses to the prover are (y, σ, r, s) such that $V \leftarrow yh^r$ and $\delta \leftarrow \sigma h^s$ (BGN commitment [10]) and $e(g^m y, \sigma) = e(g, g)$ (i.e., σ is the valid signature on m in the composite order BGN group). The NIWI proof is $\pi_0 \leftarrow g^m y^s V^r$. The verifier checks if $e(g^m y, \delta) = e(g, g) \cdot e(h, \pi_0)$ and accepts iff it holds. The completeness property easily follows. The NIWI proof is perfectly sound if h is of order q, and perfectly witness-indistinguishable if h is of order N.

NIWI proof for $\mathcal{L}_n^1$ [14]. Given a BGN bilinear group $(N, p, q, \mathbb{G}, \mathbb{G}_T, e, g)$, we define a language $\mathcal{L}_n^1 := \{(V, \{Y_i\}_n^1) | \exists i \in [n][V_p = (Y_i)_p]\}$. A NIWI proof for this language of size $\mathcal{O}(\sqrt{n})$ is given in [14]. If h has order N it is perfectly witness-indistinguishable; if h has order q it is perfectly sound. We use this tool in a black-box manner and refer the reader to [14, 28] for details. We also use a special case of the language $\mathcal{L}_2^1$ to convert a NIWI proof for $\mathcal{L}_{BB}$ to a NIZK proof for the same language.

NIZK proof for $\mathcal{L}_{BB}$. The idea is standard: add two randomly chosen groups elements Y_0 and σ_0 to the common reference string; give a NIWI proof for $\mathcal{L}_{BB}$; and use the witness of Y_0 and σ_0 to simulate the proof. Specifically, given a BGN group, to generate a NIZK proof that $(Y, m, \sigma) \in \mathcal{L}_{BB}$, one selects $r, s \overset{\$}{\leftarrow} Z_N$ and computes $V \leftarrow Yh^r$ and $\delta \leftarrow \sigma h^s$, and it then produces a NIWI proof that $(V, m, \delta) \in \mathcal{L}_{BB}$, a NIWI proof that $(V, (Y_0, Y)) \in \mathcal{L}_2^1$, and a NIWI proof that $(\delta, (\sigma, \sigma_0)) \in \mathcal{L}_2^1$.

We are ready to present the underlying verifiable random function, as depicted in Figure 4. Under the PR-DDHI assumption, it is straightforward that the unique identifier τ is pseudorandom. As in [28], $\hat{\tau}$ is used to make sure that the signer uniquely generates its signature. (Otherwise, there exists an adversary that can use them to solve the subgroup decision assumption.) We immediately have the following lemma:

Lemma 1 *The scheme presented above is a PRF with a NIZK proof under the subgroup decision assumption and the PR-DDHI assumption in both $\mathbb{G}_p$ and $\mathbb{G}_q$.* ▮

6.2 Sublinear Unique Ring Signature in Common Reference String Model

We now give our unique ring signature scheme, detailed in Figure 5, which achieves sublinear size in the common reference string model. The following theorem establishes the security of the scheme:

Theorem 3 *The above scheme presented in this section is a unique ring signature under the subgroup decision assumption and the PR-DDHI assumption in both $\mathbb{G}_p$ and $\mathbb{G}_q$.* ▮

7 Concluding Remarks

We define unique ring signatures that capture the spirit of linkable ring signature. One may regard unique ring signatures as being functionally the same as linkable ring signatures, but definitionally more simple, and more suitable for our applications. Of course, it is safe to compare the constructions between unique ring signatures and linkable ring signatures in terms of efficiency, cryptographic assumptions, and security reductions.

$\mathsf{Setup}(1^\lambda)$.

The setup algorithm serves to generate the common reference string given the security parameter λ. It first generates a BGN bilinear group of composite order $(N, p, q, \mathbb{G}, \mathbb{G}_T, e, g)$ where g is a random generator of $\mathbb{G}$ and $N = pq$. It also selects h, $\hat{h}$, Y_0, $\tau_0 \xleftarrow{\$} \mathbb{G}$, where h is used for the commitment scheme, $\hat{h}$ is used to ensure the uniqueness of the identifier, Y_0 and τ_0 is used to convert NIWI proof for $\mathcal{L}_{BB}$ to NIZK proof. It finally outputs the common reference string as

$$\mathsf{crs} = (N, \mathbb{G}, \mathbb{G}_T, e, g, h, \hat{h}, Y_0, \tau_0).$$

$\mathsf{Gen}(1^\lambda, \mathsf{crs})$.

The key generation algorithm for user i takes as input the common reference string crs and selects $x, t \xleftarrow{\$} \mathbb{Z}_N$ and computes $Y \leftarrow g^x h^t$ where we let y denote g^x. It outputs the public key as $pk = (\mathsf{crs}, Y)$ and the secret key as $sk = (\mathsf{crs}, x, t)$.

$\mathsf{Sig}(sk_i, m)$.

To sign the message m, the signer i uses its secret key x to produce a signature as

$$(m, \sigma),$$

where $\sigma = (\tau, \hat{\tau}, \pi)$ in which $\tau \leftarrow g^{\frac{1}{x+m}}$ is the unique identifier, $\hat{\tau} \leftarrow \hat{h}^{\frac{1}{x+m}}$, and π is a NIZK proof that $(Y, m, \tau) \in \mathcal{L}_{BB}$. (Note that we use Y_0 and τ_0 to convert the Groth-Sahai NIWI proof for $\mathcal{L}_{BB}$ to a NIZK proof.)

$\mathsf{Vrf}(pk_i, m, \sigma)$.

The verification algorithm first parses σ as $(\tau, \hat{\tau}, \pi)$ and verifies it by checking if π is a correct proof $\mathcal{L}_{BB}$ and $e(\tau, \hat{h}) = e(\hat{\tau}, g)$.

Figure 4: **Signature/VRF from the PR-DDHI assumption in the common reference string model**. The algorithms are described in the context of digital signature. It is also a VRF scheme, where $\mathcal{VRF}.\mathsf{Eva}(sk, m) = \tau$, $\mathcal{VRF}.\mathsf{Prove}(sk, m) = (\hat{\tau}, \pi)$, and $\mathcal{VRF}.\mathsf{Ver}(m, \sigma) = \mathcal{DS}.\mathsf{Vrf}(m, \sigma)$.

We present a general, simple, and unified framework for unique ring signature. It can be viewed as an extension and generalization of the Bellare-Goldwasser signature [4], combining certified PRF and NIZK proof of membership.

Compared to prior linkable signatures [1, 19, 37, 38, 48, 49], security of the first instantiation can be more tightly related to the simple and well-studied CDH and DDH problems in the random oracle model.

We also show how to obtain a unique ring signature scheme from the traceable ring signature due to Fujisaki [28]. Our scheme is not simply a weakened version of [28] that removes the extra public tracing functionality, but a meaningful simplification, eliminating the relatively inefficient one-time signature, employing a solo assumption, and requiring no proofs any more. (Despite its sublinear size, the scheme, however, relies on very strong assumptions and a common reference string setup, and is not as computationally efficient.)

Acknowledgments

The authors would like to thank Tsz Hon Yuen for helpful comments, and Leonid Reyzin and Sharon Goldberg for pointing out important errors on the uniqueness and unforgeability.

References

[1] M. Au, S. Chow, W. Susilo, and P. Tsang. Short linkable ring signatures revisited. *EUROPKI 2006*, LNCS vol. 4043, Springer, pp. 101–115, 2006.

$\mathsf{Setup}(1^\lambda)$.

The setup algorithm serves to generate the common reference string given the security parameter λ. It first generates a BGN bilinear group of composite order $(N, p, q, \mathbb{G}, \mathbb{G}_T, e, g)$ where g is a random generator of $\mathbb{G}$ and $N = pq$. It also selects h, $\hat{h}$, Y_0, $\tau_0 \xleftarrow{\$} \mathbb{G}$, where h is used for the commitment scheme, $\hat{h}$ is used to ensure the uniqueness of the identifier, and Y_0 and τ_0 is used to transfer NIWI proof to NIZK proof for some language $\mathcal{L}$. It then selects a collision resistant hash function $H\colon \{0,1\}^* \to \mathbb{Z}_N$. It finally outputs the common reference string as

$$\mathsf{crs} = (N, \mathbb{G}, \mathbb{G}_T, e, g, h, \hat{h}, Y_0, \tau_0).$$

$\mathsf{RG}(1^\lambda, \mathsf{crs})$.

The key generation algorithm for user i ($i \geq 1$) takes as input the common reference string crs and selects $x_i, t_i \xleftarrow{\$} \mathbb{Z}_N$ and computes $Y_i \leftarrow g^{x_i} h^{t_i}$ where we denote g^{x_i} as y_i. It outputs the public key as $pk_i = (\mathsf{crs}, Y_i)$ and the secret key as $sk_i = (\mathsf{crs}, x_i, t_i)$.

$\mathsf{RS}(sk_i, R, m)$.

To sign the message m with respect to the ring $R = (pk_1, ..., pk_n)$, the signer i uses its secret key x_i to produce a signature as

$$(R, m, \sigma),$$

where $\sigma = (\tau, \hat{\tau}, \pi)$ in which $\tau \leftarrow g^{1/(x_i + H(m||R))}$ is the unique identifier, $\hat{\tau} \leftarrow \hat{h}^{1/(x_i + H(m||R))}$, and π is a NIZK proof for the language $\mathcal{L} := \{(\{Y_j\}_1^n, m, R, \tau) | \exists j[(Y_j, H(m||R), \tau) \in \mathcal{L}_{BB}]\}$. Specifically, select $r, s \xleftarrow{\$} \mathbb{Z}_N$ and compute $V \leftarrow y_i h^r$ and $\delta \leftarrow \tau h^s$ and output V and δ; output a NIWI proof that $(V, H(m||R), \delta) \in \mathcal{L}_{BB}$; output a NIWI proof that $(V, \{Y_j\}_0^n) \in \mathcal{L}_{n+1}^1$; finally, output a NIWI proof that $(\delta, (\tau_0, \tau)) \in \mathcal{L}_2^1$.

$\mathsf{RV}(R, m, \sigma)$.

The verification algorithm first parses σ as $(\tau, \hat{\tau}, \pi)$ and checks if π is a correct proof for language $\mathcal{L}$ and $e(\tau, \hat{h}) = e(\hat{\tau}, g)$.

Figure 5: **Sublinear unique ring signature in the common reference string model**.

[2] M. Bellare, A. Boldyreva, and S. Micali. Public-key encryption in a multi-user setting: security proofs and improvements. *EUROCRYPT 2000*, LNCS vol. 1807, Springer, pp. 259–274, 2000.

[3] M. Bellare, A. Boldyreva and A. O'Neill. Deterministic and efficiently searchable encryption. *CRYPTO 2007*, LNCS vol. 4622, Springer, pp. 535-552, 2007.

[4] M. Bellare and S. Goldwasser. New paradigms for digital signatures and message authentication based on non-interative zero knowledge proofs. *CRYPTO '89*, LNCS vol. 435, Springer, pp. 194–211, 1990.

[5] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. *CCS 1993*, ACM Press, pp. 62–73, 1993.

[6] M. Bellare and P. Rogaway. The exact security of digital signatures — how to sign with RSA and Rabin. *EUROCRYPTO '96*, LNCS vol. 1070, Springer, pp. 399–416, 1996.

[7] A. Bender, J. Katz, and R. Morselli. Ring signatures: stronger definitions, and constructions without random oracles. *Journal of Cryptology* 22(1): 114–138, 2009.

[8] M. Blum, P. Feldman, and S. Micali. Non-interactive zero-knowledge and its applications. *STOC 1988*, ACM press, pp. 103–112, 1988.

[9] D. Boneh and X. Boyen. Short signatures without random oracles. *EUROCRYPT 2004*, LNCS vol. 3027, Springer, pp. 56–73, 2004.

[10] D. Boneh, E. Goh, and K. Nissim. Evaluating 2-DNF formulas on ciphertexts. *TCC '05*, LNCS vol. 3378, Springer, pp. 325–341, 2005.

[11] X. Boyen and B. Waters. Compact group signatures without random oracles. *EUROCRYPT 2006*, LNCS vol. 4004, Springer, pp. 427–444, 2006.

[12] X. Boyen and B. Waters. Full-domain subgroup hiding and constant-size group signatures. *PKC 2007*, LNCS vol. 4450, Springer, pp. 1–15, 2007.

[13] J. Camenisch, S. Hohenberger, M. Kohlweiss, A. Lysyanskaya, and M. Meyerovich. How to win the clone wars: efficient periodic n-times anonymous authentication. *CCS 2006*, ACM, pp. 201–210, 2006.

[14] N. Chandran, J. Groth, and A. Sahai. Ring signatures of sub-linear size without random oracles. *ICALP 2007*, LNCS vol. 4596, Springer, pp. 423–434, 2007.

[15] D. Chaum and H. Antwerpen. Undeniable signatures. In *CRYPTO '89*, LNCS vol. 435, Springer, pp. 212–216, 1990.

[16] D. Chaum and T. Pedersen. Wallet databases with observers. *CRYPTO '92*, LNCS vol. 740, Springer, pp. 89–105, 1993.

[17] B. Chevallier-Mames. An efficient CDH-based signature scheme with a tight security reduction. *CRYPTO 2005*, LNCS vol. 3621, Springer, pp. 511–526, 2005.

[18] S. Chow, J. Liu, and D. Wong. Robust receipt-free election system with ballot secrecy and verifiability. *NDSS 2008*, The Internet Society, 2008.

[19] S. Chow, W. Susilo, and T.H. Yuen. Escrowed linkability of ring signatures and its applications. *VIETCRYPT 2006*, LNCS vol. 4341, pp. 172–192, Springer, 2006.

[20] J. Coron. On the exact security of full-domain hash. In *CRYPTO 2000*, LNCS vol. 1880, Springer, pp. 229–235, 2000.

[21] R. Cramer, I. Damgård, and B. Schoemakers. Proofs of partial knowledge and simplified design of witness hiding protocols. *CRYPTO '94*, LNCS vol. 839, Springer, pp. 174–187, 1994.

[22] I. Damgård, K. Dupont, and M. Pedersen. Unclonable group identification. *EUROCRYPT 2006*, LNCS vol. 4004, Springer, pp. 555–572, 2006.

[23] U. Feige, D. Lapidot, and A. Shamir. Multiple non-interactive zero knowledge proofs under general assumptions. *SIAM J. Computing*, 29(1):1–28, 1999.

[24] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. *CRYPTO '86*, LNCS vol. 263, Springer, pp. 186–194, 1987.

[25] M. Fischlin. Communication-efficient non-interactive proofs of knowledge with online extractors. *CRYPTO 2005*, LNCS vol. 3621, Springer, pp. 152–168, 2005.

[26] M. Franklin and H. Zhang. Unique group signatures. *ESORICS 2012*. Full version in Cryptology ePrint Archive: Report 2012/204. http://eprint.iacr.org

[27] E. Fujisaki and K. Suzuki. Traceable ring signature. *IEICE Transactions 91-A(1)*: 83–93 (2008).

[28] E. Fujisaki. Sub-linear size traceable ring signatures without random oracles. *CT-RSA '11*, LNCS vol. 6558, Springer, pp. 393–415, 2011.

[29] E. Goh and S. Jarecki. A signature scheme as secure as the Diffie-Hellman problem. *EUROCRYPT 2003*, LNCS vol. 2656, Springer, pp. 401–415, 2003.

[30] E. Goh, S. Jarecki, J. Katz and N. Wang. Efficient Signature Schemes with Tight Security Reductions to the Diffie-Hellman Problems. *Journal of Cryptology* 20(4): 493–514, 2007.

[31] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, 1986.

[32] S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[33] J. Groth, R. Ostrovsky, and A. Sahai. Perfect non-interactive zeroknowledge for NP. *EUROCRYPT 2006*, LNCS vol. 4004, Springer, pp. 339–358, 2006.

[34] J. Groth, A. Sahai. Efficient non-interactive proof systems for bilinear groups. *EUROCRYPT 2008*, LNCS vol. 4965, Springer, pp. 415–432, 2008.

[35] J. Herranz and G. Sáez. Forking lemmas for ring signature schemes. *INDOCRYPT 2003*, LNCS vol. 2904, Springer, pp. 266–279, 2003.

[36] J. Katz and N. Wang. Efficiency improvements for signature schemes with tight security reductions. *CCS 2003*, ACM press, pp. 155–164, 2003.

[37] J. Liu, V. Wei, and D. Wong. Linkable spontaneous anonymous group signatures for ad hoc groups. *ACISP 2004*, LNCS vol. 3108, Springer, pp. 325–335, 2004.

[38] J. Liu and D. Wong. Linkable ring signatures: Security models and new schemes. *ICCSA 2005*, LNCS vol. 3481, Springer, pp. 614–623, 2005.

[39] S. Micali, M. Rabin, and S. Vadhan. Verifiable random functions. *FOCS 1999*, IEEE Computer Society, pp. 120–130, 1999.

[40] S. Micali and L. Reyzin. Improving the exact security of digital signature schemes. *J. Cryptology*, 15(1): 1–18, 2002.

[41] L. Nguyen and R. Safavi-Naini. Dynamic k-Times anonymous authentication. *ACNS 2005*, LNCS

vol. 3531, Springer, pp. 219–250, 2005.

[42] D. Papadopoulos, D. Wessels, S. Huque, J. Včelák, M. Naor, L. Reyzin, and S. Goldberg. Can NSEC5 be practical for DNSSEC deployments? Cryptology ePrint Archive, Report 2017/099, 2017. http://eprint.iacr.org.

[43] D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *J. Cryptology*, 13(3): 361–396, 2000.

[44] R. Rivest, A. Shamir, and Y. Tauman. How to leak a secret: Theory and applications of ring signatures. *Theoretical Computer Science, Essays in Memory of Shimon Even*, LNCS vol. 3895, Springer, pp. 164–186, 2006.

[45] C.-P. Schnorr. Efficient identification and signatures for smart cards. *CRYPTO '89*, LNCS vol. 435, Springer, pp. 239–252, 1990.

[46] I. Teranishi, J. Furukawa, and K. Sako. k-times anonymous authentication. *ASIACRYPT 2004*, LNCS vol. 3329, Springer, pp. 308–322, 2004.

[47] I. Teranishi, K. Sako. k-times anonymous authentication with a constant proving cost. *PKC 2006*, LNCS vol. 3958, Springer, pp. 525–542, 2006.

[48] P. Tsang and V. Wei. Short linkable ring signatures for e-voting, e-cash and attestation. *IPSEC 2005*, LNCS vol. 3439, Springer, pp. 48–60, 2005.

[49] P. Tsang, V. Wei, T. Chan, M. Au, J. Liu, and D. Wong. Separable linkable threshold ring signatures. *INDOCRYPT 2004*, LNCS vol. 3348, Springer, pp. 389–398, 2004.