Group Signatures with Verifier-Local Revocation and Backward Unlinkability in the Standard Model

Benoît Libert, Damien Vergnaud

Group Signatures with Verifier-Local Revocation and Backward Unlinkability in the Standard Model

Benoît Libert[1] and Damien Vergnaud[2] *

[1] Université Catholique de Louvain, Microelectronics Laboratory, Crypto Group
Place du Levant, 3 – 1348 Louvain-la-Neuve – Belgium
[2] École normale supérieure – C.N.R.S. – I.N.R.I.A.
45, Rue d'Ulm – 75230 Paris CEDEX 05 – France

Abstract. Group signatures allow users to anonymously sign messages in the name of a group. Membership revocation has always been a critical issue in such systems. In 2004, Boneh and Shacham formalized the concept of group signatures with *verifier-local revocation* where revocation messages are only sent to signature verifiers (as opposed to both signers and verifiers). This paper presents an efficient verifier-local revocation group signature (VLR-GS) providing *backward unlinkability* (*i.e.* previously issued signatures remain anonymous even after the signer's revocation) with a security proof in the standard model (*i.e.* without resorting to the random oracle heuristic).

Keywords. Group signatures, verifier-local revocation, bilinear maps, backward unlinkability, standard model.

1 Introduction

The *group signature* primitive, as introduced by Chaum and van Heyst in 1991 [15], allows members of a group to sign messages, while hiding their identity within a population group members administered by a group manager. At the same time, it must be possible for a tracing authority holding some trapdoor information to "open" signatures and find out which group members are their originator. A major issue in group signatures is the revocation of users whose membership should be cancelled: disabling the signing capability of misbehaving members (or honest users who intentionally leave the group) without affecting remaining members happens to be a highly non-trivial problem. In 2004, Boneh and Shacham [9] formalized the concept of group signatures with *verifier-local revocation* where revocation messages are only sent to signature verifiers (as opposed to both signers and verifiers). This paper describes the first efficient verifier-local revocation group signature scheme providing *backward unlinkability* (*i.e.*, previously issued signatures remain anonymous even after the signer's revocation) whose proof of security does not hinge upon the random oracle heuristic.

1.1 Related Work

Group signatures. Many group signatures were proposed in the nineties, the first provably coalition-resistant proposal being the famous ACJT scheme [2] proposed by Ateniese, Camenisch, Joye and Tsudik in 2000. The last few years saw the appearance of new constructions using bilinear maps [7, 33, 20, 17]. Among these, the Boneh-Boyen-Shacham scheme [7] was the first one to offer signatures shorter than 200 bytes using the *Strong Diffie-Hellman assumption* [6]. Its security was analyzed using random oracles [4] in the model of Bellare, Micciancio and Warinschi [3] (BMW) which captures all the requirements of group signatures in three well-defined properties.

* The first author acknowledges the Belgian National Fund for Scientific Research (F.R.S.-F.N.R.S.) for their financial support and the BCRYPT Interuniversity Attraction Pole. The second author is supported by the European Commission through the ICT Program under Contract ICT-2007-216676 ECRYPT II and by the French *Agence Nationale de la Recherche* through the ANR 07-TCOM-013-04 PACE Project.

The BMW model, which assumes static groups where no new member can be introduced after the setup phase, was independently extended by Kiayias and Yung [27] and Bellare-Shi-Zhang [5] to a dynamic setting. In these models (that are very close to each other), efficient pairing-based schemes were put forth by Nguyen and Safavi-Naini [33], Furukawa and Imai [20] and, later on, by Delerablée and Pointcheval [17]. In dynamically growing groups, Ateniese *et al.* [1] also proposed a construction without random oracles offering a competitive efficiency at the expense of a security resting on interactive assumptions that are not efficiently falsifiable [32]. Another standard model proposal was put forth (and subsequently improved [11]) by Boyen-Waters [10] in the static model from [3] under more classical assumptions. Groth [22] described a scheme with constant-size signatures without random oracles in the dynamic model [5] but signatures were still too long for practical use. Later on, Groth showed [23] a fairly practical random-oracle-free group signature with signature length smaller than 2 kB and full anonymity (*i.e.*, anonymity in a model where the adversary is allowed to open anonymous signatures at will) in the model of [5].

Verifier-local revocation. Membership revocation has always been a critical issue in group signatures. The simplest solution is to generate a new group public key and provide unrevoked signers with a new signing key, which implies the group master to send a secret message to each individual signer as well as to broadcast a public message to verifiers. In some settings, it may not be convenient to send a new secret to signers after their inclusion in the group. In *verifier-local revocation group signatures* (VLR-GS), originally suggested in [14] and formalized in [9], revocation messages are only sent to verifiers (making the group public key and the signing procedure independent of which and how many members were excluded). The group manager maintains a (periodically updated) revocation list (RL) which is used by all verifiers to perform the revocation test and make sure that signatures were not produced by a revoked member.

The RL contains a token for each revoked user. The verification algorithm accepts all signatures issued by unrevoked users and reveals no information about which unrevoked user issued the signature. However, if a user is revoked, his signatures are no longer accepted. It follows that signatures from a revoked member become linkable: to test that two signatures emanate from the same revoked user, one can simply verify signatures once using the RL before the alleged signer's revocation and once using the post-revocation RL. As a result, users who deliberately leave the group inevitably lose their privacy.

The property of *backward unlinkability*, first introduced in [35] in the context of key-evolving group signatures, ensures that signatures that were generated by a revoked member *before* his revocation remain anonymous and unlinkable. This property is useful when members who voluntarily leave the group wish to retain a certain level of privacy. When users' private keys get stolen, preserving the anonymity of their prior signatures is also definitely desirable.

Boneh and Shacham [9] proposed a VLR group signature using bilinear maps in a model inspired from [3]. In [30], Nakanishi and Funabiki extended Boneh-Shacham group signatures and devised a scheme providing backward unlinkability. They proved the anonymity of their construction under the Decision Bilinear Diffie-Hellman assumption [8]. In [31], the same authors suggested another backward-unlinkable scheme with shorter signatures. Other pairing-based VLR-GS constructions were put forth in [38, 39]

Traceable signatures [26], that also have pairing-based realizations [33, 16], can be seen as extensions of VLR-GS schemes as they also admit an implicit tracing mechanism. They provide additional useful properties such as the ability for signers to claim (and prove) the authorship of anonymously generated signatures or the ability for the group manager to reveal a trapdoor allowing to publicly trace all signatures created by a given user. This primitive was recently implemented in the standard model [29]. However, it currently does not provide a way to trace users' signatures per period: once the tracing trapdoor of some group member is revealed, all

signatures created by that member become linkable. In some situations, it may be desirable to obtain a fine-grained traceability and only trace signatures that were issued in specific periods. The problem of VLR-GS schemes with backward unlinkability can be seen as the one of tracing some user's signatures from a given period onwards while preserving the anonymity and the unlinkability of that user's signatures for earlier periods. The solution described in this paper readily extends to retain the anonymity of signatures produced during past *and* future periods.

1.2 Contribution of the paper.

All known constructions of group signatures with verifier local revocation (with or without backward unlinkability) make use of the Fiat-Shamir paradigm [18] and thus rely on the random oracle methodology [4], which is known not to provide more than heuristic arguments in terms of security. Failures of the random oracle model were indeed reported in several papers such as [13, 21]. When first analyzed in the random oracle model, cryptographic primitives thus deserve further efforts towards securely instantiating them without appealing to the random oracle idealization.

The contribution of this paper is to describe a new VLR-GS scheme with backward unlinkability in the standard model. Recently, Groth and Sahai [24] described powerful non-interactive proof systems allowing to prove that a number of committed variables satisfy certain algebraic relations. Their techniques notably proved useful to design standard model group signatures featuring constant signature size [11, 22, 23].

Extending the aforementioned constructions to obtain VLR-GS schemes with backward unlinkability is not straightforward. The approach used in [31], which can be traced back to Boneh-Shacham [9], inherently requires to use programmable random oracles, the behavior of which currently seems impossible to emulate in the standard model (even with the techniques developed in [25]). Another approach used in [30] looks more promising as it permits traceability with backward unlinkablity without introducing additional random oracles. This technique, however, does not interact with the Groth-Sahai toolbox in a straightforward manner as it typically requires non-interactive zero-knowledge (NIZK) proofs for what Groth and Sahai called *pairing product equations*. The problem that we face is that proving the required anonymity property of VLR-GS schemes entails to simulate a NIZK proof for such a pairing-product equation at some step of the reduction. As pointed out in [24], such non-interactive proofs are only known to be simulatable in NIZK under specific circumstances that are not met if we try to directly apply the technique of [30].

To address the above technical difficulty, we use the same revocation mechanism as [30] but use a slightly stronger (but still falsifiable [32]) assumption in the proof of anonymity: while Nakanishi and Funabiki rely the Decision Bilinear Diffie-Hellman assumption, we rest on the hardness of the so-called Decision Tripartite Diffie-Hellman problem, which is to distinguish g^{abc} from random given (g, g^a, g^b, g^c). Our contribution can be summarized as showing that the implicit tracing mechanism of [30] can be safely applied to the Boyen-Waters group signature [11] to make it backward-unlinkably revocable. This property comes at the expense of a quite moderate increase of signature sizes w.r.t. [11]. The main price to pay is actually to use a slightly stronger assumption than in [30] in the security proof.

2 Preliminaries

2.1 Verifier-Local Revocation Group Signatures

This section presents the model of VLR group signatures with backward unlinkability proposed in [30] which extends the Boneh-Shacham model [9] of VLR group signatures.

Definition 1. *A VLR group signature scheme with backward unlinkability consists of the following algorithms:*

Keygen(λ, N, T)*: is a randomized algorithm taking as input a security parameter* $\lambda \in \mathbb{N}$ *and integers* $N, T \in \mathbb{N}$ *indicating the number of group members and the number of time periods, respectively.*
Its output consists of a group public key gpk*, a* N*-vector of group members' secret keys* $\mathsf{gsk} = (\mathsf{gsk}[1], \ldots, \mathsf{gsk}[N])$ *and a* $(N \times T)$*-vector of revocation tokens* $\mathsf{grt} = (\mathsf{grt}[1][1], \ldots, \mathsf{grt}[N][T])$*, where* $\mathsf{grt}[i][j]$ *indicates the token of member* i *at time interval* j*.*

Sign$(\mathsf{gpk}, \mathsf{gsk}[i], j, M)$ *: is a possibly randomized algorithm taking as input, the group public key* gpk*, the current time interval* j*, a group member's secret key* $\mathsf{gsk}[i]$ *and a message* $M \in \{0,1\}^*$*. It outputs a group signature* σ*.*

Verify$(\mathsf{gpk}, j, RL_j, \sigma, M)$*: is a deterministic algorithm taking as input* gpk*, the period number* j*, a set of revocation tokens* RL_j *for period* j*, a signature* σ*, and the message* M*. It outputs either "*`valid`*" or "*`invalid`*". The former output indicates that* σ *is a correct signature on* M *at interval* j *w.r.t.* gpk*, and that the signer is not revoked at interval* j*.*

For all $(\mathsf{gpk}, \mathsf{gsk}, \mathsf{grt}) = \mathbf{Keygen}(\lambda, N, T)$*, all* $j \in \{1, \ldots, T\}$*, all* RL_j*, all* $i \in \{1, \ldots, N\}$ *and any message* $M \in \{0,1\}^*$*, it is required that if* $\mathsf{grt}[i][j] \notin RL_j$ *then:*

$$\mathbf{Verify}(\mathsf{gpk}, j, RL_j, \mathbf{Sign}(\mathsf{gpk}, \mathsf{gsk}[i], j, M), M) = \text{“}\texttt{valid}\text{”}.$$

Remark 1. As mentioned in [9], any such group signature scheme has an associated *implicit tracing* algorithm that allows tracing a signature to the group member who generated it using the vector grt as the tracing key: on input a valid message-signature pair (M, σ) for period j, the opener can determine which user was the author of σ by successively executing the verification algorithm on (M, σ) using the vector of revocation tokens (*i.e.*, with $RL_j = \{\mathsf{grt}[i][j]\}_{i \in \{1,\ldots,N\}}$) and outputting the first index $i \in \{1, \ldots, N\}$ for which the verification algorithm returns "`invalid`" whereas verifying the same pair (M, σ) with $RL_j = \emptyset$ yields the answer "`valid`".

From a security standpoint, VLR group signatures with backward unlinkability should satisfy the following properties:

Definition 2. *A VLR-GS with backward unlinkability has the* **traceability** *property if no probabilistic polynomial time (PPT) adversary* $\mathcal{A}$ *has non-negligible advantage in the following game.*

1. *The challenger* $\mathcal{C}$ *runs the setup algorithm to produce a group public key* gpk*, a group master secret* gsk *and a vector* grt *of revocation tokens. It also defines a set of corrupt users* U *which is initially empty. The adversary* $\mathcal{A}$ *is provided with* gpk *and* grt *while* $\mathcal{C}$ *keeps* gsk *to itself.*
2. $\mathcal{A}$ *can make a number of invocations to the following oracles:*

   Signing oracle: *on input of a message* M*, an index* $i \in \{1, \ldots, N\}$ *and a period number* j*, this oracle responds with a signature* σ *generated on behalf of member* i *for period* j*.*

   Corruption oracle: *given an index* $i \in \{1, \ldots, N\}$*, this oracle reveals the private key* $\mathsf{gsk}[i]$ *of member* i *which is included in the set* U*.*

3. $\mathcal{A}$ *eventually comes up with a signature* $\sigma^\star$ *on a message* $M^\star$*, a period number* $j^\star$ *and a set of revocation tokens* $RL^\star_{j^\star}$*.*

The adversary $\mathcal{A}$ *is declared successful if*

– $\mathbf{Verify}(\mathsf{gpk}, j^\star, RL^\star_{j^\star}, \sigma^\star, M^\star) = \text{“}\texttt{valid}\text{”}$.

– *The execution of the implicit tracing algorithm on input of the vector of revocation tokens* $(\mathsf{grt}[1][j^\star], \ldots, \mathsf{grt}[N][j^\star])$*, ends up in one of the following ways:*
  - $\sigma^\star$ *traces to a member outside the coalition* $U \backslash RL^\star_{j^\star}$ *that did not sign* $M^\star$ *during period* $j^\star$
  - *the tracing fails.*

$\mathcal{A}$*'s advantage in breaking traceability is measured as*

$$\mathbf{Adv}^{\text{trace}}_{\mathcal{A}}(k) \;:=\; \Pr[\mathcal{A} \textit{ is successful}],$$

where the probability is taken over the coin tosses of $\mathcal{A}$ *and the challenger.*

This definition slightly weakens the original one [30] that captures the strong unforgeability requirement (*i.e.*, the message-signature pair $(M^\star, \sigma^\star)$ must be different from that of any signing query during period $j^\star$). Due to the use of publicly randomizable non-interactive witness indistinguishable proofs, we need to settle for the usual flavor of unforgeability according to which the message $M^\star$ must not have been queried for signature during the target period $j^\star$.

Definition 3. *A VLR-GS with backward unlinkability provides* **BU-anonymity** *if no PPT adversary* $\mathcal{A}$ *has non-negligible advantage in the following game.*

1. *The challenger* $\mathcal{C}$ *runs* $\mathbf{Keygen}(\lambda, n, T)$ *to produce a group public key* gpk*, a master secret* gsk *and a vector* grt *of revocation tokens. The adversary* $\mathcal{A}$ *is given* gpk *but is denied access to* grt *and* gsk*.*
2. *At the beginning of each period,* $\mathcal{C}$ *increments a counter* j *and notifies* $\mathcal{A}$ *about it. During the current time interval* j*,* $\mathcal{A}$ *can adaptively invoke the following oracles:*

   Signing oracle: *on input of a message* M *and an index* $i \in \{1, \ldots, n\}$*, this oracle outputs a signature* σ *generated for member* i *and period* j*.*

   Corruption oracle: *for an adversarially-chosen* $i \in \{1, \ldots, n\}$*, this oracle reveals member* i*'s private key* $\mathsf{gsk}[i]$*.*

   Revocation oracle: *given* $i \in \{1, \ldots, n\}$*, this oracle outputs member* i*'s revocation token for the current period* j*.*
3. *At some period* $j^\star \in \{1, \ldots, T\}$*,* $\mathcal{A}$ *comes up with a message* M *and two distinct user indices* $i_0, i_1 \in \{1, \ldots, n\}$ *such that neither* i_0 *or* i_1 *has been corrupt. Moreover, they cannot have been revoked before or during period* $j^\star$*. At this stage,* $\mathcal{C}$ *flips a fair coin* $d^\star \xleftarrow{R} \{0,1\}$ *and generates a signature* $\sigma^\star$ *on* M *on behalf of user* $i_{d^\star}$ *which is sent as a challenge to* $\mathcal{A}$*.*
4. $\mathcal{A}$ *is granted further oracle accesses as in phase 2. Of course, she may not query the private key of members* i_0, i_1 *at any time. On the other hand, she may obtain their revocation tokens for time intervals after* $j^\star$*.*
5. *Eventually,* $\mathcal{A}$ *outputs* $d' \in \{0,1\}$ *and wins if* $d' = d^\star$*.*

The advantage of $\mathcal{A}$ *in breaking BU-anonymity is defined as* $\mathbf{Adv}^{\text{bu-anon}}_{\mathcal{A}}(k) \;:=\; |\Pr[d' = d^\star] - 1/2|$*, where the probability is taken over all coin tosses.*

2.2 Bilinear Maps and Complexity Assumptions

Bilinear groups. Groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order p are called *bilinear groups* if there is an efficiently computable mapping $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ such that:

1. $e(g^a, h^b) = e(g, h)^{ab}$ for any $(g, h) \in \mathbb{G} \times \mathbb{G}$ and $a, b \in \mathbb{Z}$;
2. $e(g, h) \neq 1_{\mathbb{G}_T}$ whenever $g, h \neq 1_{\mathbb{G}}$.

In such groups, we will need three non-interactive (and thus falsifiable [32]) complexity assumptions.

Definition 4. *In a group $\mathbb{G} = \langle g \rangle$ of prime order $p > 2^\lambda$, the* **Decision Linear Problem** *(DLIN) is to distinguish the distributions $(g, g^a, g^b, g^{ac}, g^{bd}, g^{c+d})$ and $(g, g^a, g^b, g^{ac}, g^{bd}, g^z)$, with $a, b, c, d \xleftarrow{R} \mathbb{Z}_p^*$, $z \xleftarrow{R} \mathbb{Z}_p^*$. The* **Decision Linear Assumption** *posits that, for any PPT distinguisher $\mathcal{D}$,*

$$\mathbf{Adv}^{\mathrm{DLIN}}_{\mathbb{G},\mathcal{D}}(\lambda) = |\Pr[\mathcal{D}(g, g^a, g^b, g^{ac}, g^{bd}, g^{c+d}) = 1 | a, b, c, d \xleftarrow{R} \mathbb{Z}_p^*] - \Pr[\mathcal{D}(g, g^a, g^b, g^{ac}, g^{bd}, g^z) = 1 | a, b, c, d \xleftarrow{R} \mathbb{Z}_p^*,\ z \xleftarrow{R} \mathbb{Z}_p^*]| \in \mathsf{negl}(\lambda).$$

This problem amounts to deciding whether vectors $\vec{g_1} = (g^a, 1, g)$, $\vec{g_2} = (1, g^b, g)$ and $\vec{g_3}$ are linearly dependent or not. It has been used [24] to construct efficient non-interactive proof systems.

We also rely on a variant, introduced by Boyen and Waters [11], of the Strong Diffie-Hellman assumption [6].

Definition 5 ([11]). *In a group $\mathbb{G}$ of prime order p, the* **ℓ-Hidden Strong Diffie-Hellman problem** *(ℓ-HSDH) is, given elements $(g, \Omega = g^\omega, u) \xleftarrow{R} \mathbb{G}^3$ and ℓ distinct triples $(g^{1/(\omega+s_i)}, g^{s_i}, u^{s_i})$ with $s_1, \ldots, s_\ell \xleftarrow{R} \mathbb{Z}_p^*$, to find another triple $(g^{1/(\omega+s)}, g^s, u^s)$ such that $s \neq s_i$ for $i \in \{1, \ldots, \ell\}$.*

We also rely on the following intractability assumption suggested for the first time in [8, Section 8].

Definition 6. *In a prime order group $\mathbb{G}$, the* ***Decision Tripartite Diffie-Hellman Assumption*** *(DTDH) is the infeasibility of deciding if $\eta = g^{abc}$ on input of (g, g^a, g^b, g^c, η), where $a, b, c \xleftarrow{R} \mathbb{Z}_p^*$. The advantage function $\mathbf{Adv}^{\mathrm{DTDH}}_{\mathbb{G},\mathcal{D}}(\lambda)$ of any PPT distinguisher $\mathcal{D}$ is defined analogously to the DLIN case.*

The above assumption is a bit stronger than the widely accepted Decision Bilinear Diffie-Hellman assumption according to which the distributions

$$\{(g, g^a, g^b, g^c, e(g,g)^{abc}) | a, b, c, \xleftarrow{R} \mathbb{Z}_p\} \text{ and } \{(g, g^a, g^b, g^c, e(g,g)^z) | a, b, c, z \xleftarrow{R} \mathbb{Z}_p\}$$

are computationally indistinguishable. Yet, the DTDH problem is still believed to be hard in groups with a bilinear map where the DDH problem is easy.

2.3 Groth-Sahai Proof Systems

In the following notations, for equal-dimension vectors or matrices A and B containing group elements, $A \odot B$ stands for their entry-wise product (*i.e.* it denotes their Hadamard product).

When based on the DLIN assumption, the Groth-Sahai (GS) proof systems [24] use a common reference string comprising vectors $\vec{g_1}, \vec{g_2}, \vec{g_3} \in \mathbb{G}^3$, where $\vec{g_1} = (g_1, 1, g)$, $\vec{g_2} = (1, g_2, g)$ for some $g_1, g_2 \in \mathbb{G}$. To commit to group elements $X \in \mathbb{G}$, one sets $\vec{C} = (1, 1, X) \odot \vec{g_1}^r \odot \vec{g_2}^s \odot \vec{g_3}^t$ with $r, s, t \xleftarrow{R} \mathbb{Z}_p^*$. When the proof system is configured to give perfectly sound proofs, $\vec{g_3}$ is chosen as $\vec{g_3} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2}$ with $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$. Commitments $\vec{C} = (g_1^{r+\xi_1 t}, g_2^{s+\xi_2 t}, X \cdot g^{r+s+t(\xi_1+\xi_2)})$ are then Boneh-Boyen-Shacham (BonehBoyenShacham2004) ciphertexts that can be decrypted using $\alpha_1 = \log_g(g_1)$, $\alpha_2 = \log_g(g_2)$. In the witness indistinguishability (WI) setting, vectors $\vec{g_1}, \vec{g_2}, \vec{g_3}$ are linearly independent and $\vec{C}$ is a perfectly hiding commitment. Under the DLIN assumption, the two kinds of CRS are computationally indistinguishable.

To commit to a scalar $x \in \mathbb{Z}_p$, one computes $\vec{C} = \vec{\varphi}^x \odot \vec{g_1}^r \odot \vec{g_2}^s$, with $r, s \xleftarrow{R} \mathbb{Z}_p^*$, using a CRS

comprising vectors $\vec{\varphi}, \vec{g_1}, \vec{g_2}$. In the soundness setting $\vec{\varphi}, \vec{g_1}, \vec{g_2}$ are linearly independent (typically $\vec{\varphi} = \vec{g_3} \odot (1, 1, g)$ where $\vec{\varphi} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2}$) whereas, in the WI setting, choosing $\vec{\varphi} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2}$ gives a perfectly hiding commitment since $\vec{C}$ is always a BonehBoyenShacham2004 encryption of $1_{\mathbb{G}}$.

To prove that committed variables satisfy a set of relations, the GS techniques replace variables by commitments in each relation. The whole proof consists of one commitment per variable and one proof element (made of a constant number of group elements) per relation.

Such proofs are easily obtained for pairing-product relations, which are of the type

$$\prod_{i=1}^{n} e(\mathcal{A}_i, \mathcal{X}_i) \cdot \prod_{i=1}^{n} \cdot \prod_{j=1}^{n} e(\mathcal{X}_i, \mathcal{X}_j)^{a_{ij}} = t_T, \tag{1}$$

for variables $\mathcal{X}_1, \ldots, \mathcal{X}_n \in \mathbb{G}$ and constants $t_T \in \mathbb{G}_T, \mathcal{A}_1, \ldots, \mathcal{A}_n \in \mathbb{G}, a_{ij} \in \mathbb{G}$, for $i, j \in \{1, \ldots, n\}$. Efficient proofs also exist for multi-exponentiation equations

$$\prod_{i=1}^{m} \mathcal{A}_i^{y_i} \cdot \prod_{j=1}^{n} \mathcal{X}_j^{b_j} \cdot \prod_{i=1}^{m} \cdot \prod_{j=1}^{n} \mathcal{X}_j^{y_i \gamma_{ij}} = T, \tag{2}$$

for variables $\mathcal{X}_1, \ldots, \mathcal{X}_n \in \mathbb{G}$, $y_1, \ldots, y_m \in \mathbb{Z}_p$ and constants $T, \mathcal{A}_1, \ldots, \mathcal{A}_m \in \mathbb{G}$, $b_1, \ldots, b_n \in \mathbb{Z}_p$ and $\gamma_{ij} \in \mathbb{G}$, for $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$.

In both cases, proofs for quadratic equations cost 9 group elements. Linear pairing-product equations (when $a_{ij} = 0$ for all i, j) take 3 group elements each. Linear multi-exponentiation equations of the type $\prod_{j=1}^{n} \mathcal{X}_j^{b_j} = T$ (resp. $\prod_{i=1}^{m} \mathcal{A}_i^{y_i} = T$) demand 3 (resp. 2) group elements.

Multi-exponentiation equations admit zero-knowledge proofs at no additional cost. On a simulated CRS (prepared for the WI setting), a trapdoor makes it is possible to simulate proofs without knowing witnesses and simulated proofs are identically distributed to real proofs.

On the other hand, pairing-product equations are not known to always have zero-knowledge proofs. Proving relations of the type (1) in NIZK usually comes at some expense since auxiliary variables have to be introduced and proof sizes are not necessarily independent of the number of variables. If $t_T = 1_{\mathbb{G}_T}$ in relation (1), the NIZK simulator can always use $\mathcal{X}_1 = \cdots = \mathcal{X}_n = 1_{\mathbb{G}}$ as witnesses. If t_T equals $\prod_{j=1}^{n'} e(g_j, h_j)$ for known group elements $g_1, \ldots, g_{n'}, h_1, \ldots, h_{n'} \in \mathbb{G}$, the simulator can prove that

$$\prod_{i=1}^{n} e(\mathcal{A}_i, \mathcal{X}_i) \cdot \prod_{i=1}^{n} \cdot \prod_{j=1}^{n} e(\mathcal{X}_i, \mathcal{X}_j)^{a_{ij}} = \prod_{j=1}^{n'} e(g_j, \mathcal{Y}_j) \tag{3}$$

and that introduced variables $\mathcal{Y}_1, \ldots, \mathcal{Y}_{n'}$ satisfy the linear equations $\mathcal{Y}_j = h_j$ for $j \in \{1, \ldots, n'\}$. Since linear equations are known to have NIZK proofs and the proof of relation (3) can be simulated using witnesses $\mathcal{X}_1 = \cdots = \mathcal{X}_n = \mathcal{Y}_1 = \cdots = \mathcal{Y}_{n'} = 1_{\mathbb{G}}$. When t_T is an arbitrary element of $\mathbb{G}_T$, pairing-product equations are currently not known to have NIZK proofs at all.

3 A Scheme in the Standard Model

3.1 Description of the scheme

In notations hereafter, it will be useful to define the coordinate-wise pairing $E : \mathbb{G} \times \mathbb{G}^3 \to \mathbb{G}_T^3$ such that, for any $h \in \mathbb{G}$ and any vector $\vec{g} = (g_1, g_2, g_3) \in \mathbb{G}^3$, $E\big(h, \vec{g}\big) = \big(e(h, g_1), e(h, g_2), e(h, g_3)\big)$. As in [24], we will also make use of a symmetric bilinear map $F : \mathbb{G}^3 \times \mathbb{G}^3 \to \mathbb{G}_T$ defined in such a way that, for any vectors $\vec{X} = (X_1, X_2, X_3) \in \mathbb{G}^3$ and $\vec{Y} = (Y_1, Y_2, Y_3) \in \mathbb{G}^3$, we have

$F(\vec{X},\vec{Y}) = \tilde{F}(\vec{X},\vec{Y})^{1/2} \cdot \tilde{F}(\vec{Y},\vec{X})^{1/2}$, where $\tilde{F} : \mathbb{G}^3 \times \mathbb{G}^3 \to \mathbb{G}_T^9$ is a non-commutative bilinear mapping that sends $(\vec{X},\vec{Y})$ onto the matrix $\tilde{F}(\vec{X},\vec{Y})$ of entry-wise pairings (*i.e.*, containing $e(X_i, Y_j)$ in its entry (i,j)).

Also, for any $z \in \mathbb{G}_T$, $\iota_T(z)$ denotes the 3×3 matrix containing z in position $(3,3)$ and 1 everywhere else. For group elements $X \in \mathbb{G}$, the notation $\iota(X)$ will denote the vector $(1,1,X) \in \mathbb{G}^3$.

The group manager holds a public key $(g, \Omega = g^\omega, A = e(g,g)^\alpha, u)$, where (α, γ) is the private key. As in the Boyen-Waters construction [11], group members' private keys consist of triples $(K_1, K_2, K_3) = \big((g^\alpha)^{1/(\omega+s_i)}, g^{s_i}, u^{s_i}\big)$, where s_i uniquely identifies the group member. Messages can be signed by creating tuples $(S_1, S_2, S_3, S_4) = (K_1, K_2, K_3 \cdot F(m)^r, g^r)$, where r is a random exponent and $F : \{0,1\}^* \to \mathbb{G}$ is a Waters-like hash function [36].

The revocation mechanism of [30] consists in introducing a vector $(h_1, \ldots, h_T)$ of group elements, where T is the number of time periods, that allow to form revocation tokens for each user: the revocation token of user i for period j is obtained as $\mathsf{grt}[i][j] = h_j^{s_i}$. When user i must be revoked at stage j, the group manager can simply add $\mathsf{grt}[i][j]$ to the revocation list RL_j of period j. When user i signs a message during stage j, he is required to include a pair $(T_1, T_2) = \big(g^\delta, e(h_j, g^{s_i})^\delta\big)$ in the signature and append a proof that $(g, T_1 = g^\delta, K_2 = g^{s_i}, h_j, T_2)$ satisfy the forementioned relation and that T_2 is indeed the "Bilinear Diffie-Hellman value" $e(h_j, g^{s_i})^\delta$ associated with (g, T_1, K_2, h_j).

Keygen(λ, N, T)**:** for security parameters λ and $n \in \mathsf{poly}(\lambda)$, choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of order $p > 2^\lambda$, with $g, h_1, \ldots, h_T, u \xleftarrow{R} \mathbb{G}$. Select $\alpha, \omega \xleftarrow{R} \mathbb{Z}_p^*$ and set $A = e(g,g)^\alpha$, $\Omega = g^\omega$. Select $\overline{v} = (v_0, v_1, \ldots, v_n) \xleftarrow{R} \mathbb{G}^{n+1}$. Choose vectors $\mathbf{g} = (\vec{g_1}, \vec{g_2}, \vec{g_3})$ such that $\vec{g_1} = (g_1, 1, g) \in \mathbb{G}^3$, $\vec{g_2} = (1, g_2, g) \in \mathbb{G}^3$, and $\vec{g_3} = \vec{g_1}^{\xi_1} \cdot \vec{g_2}^{\xi_2}$, with $g_1 = g^{\alpha_1}, g_2 = g^{\alpha_2}$ and $\alpha_1, \alpha_2 \xleftarrow{R} \mathbb{Z}_p^*$, $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p$. Finally, select a collision-resistant hash function $H : \{0,1\}^* \to \{0,1\}^n$. The group public key is defined to be

$$\mathsf{gpk} := \Big(g,\ h_1, \ldots,\ h_T,\ A = e(g,g)^\alpha,\ \Omega = g^\omega,\ u,\ \overline{v},\ \mathbf{g},\ H\Big)$$

while the group manager's private key is $\big(\alpha, \omega, \alpha_1, \alpha_2\big)$. User i is assigned the group signing key $\mathsf{gsk}[i] = (K_1, K_2, K_3) = \big((g^\alpha)^{\frac{1}{\omega+s_i}}, g^{s_i}, u^{s_i}\big)$ and his revocation token for period $j \in \{1, \ldots, T\}$ is defined as $\mathsf{grt}[i][j] := h_j^{s_i}$.

Sign$(\mathsf{gpk}, \mathsf{gsk}[i], j, M)$**:** given $\mathsf{gsk}[i] = (K_1, K_2, K_3) = \big((g^\alpha)^{\frac{1}{\omega+s_i}}, g^{s_i}, u^{s_i}\big)$, to sign a message M during period j, the signer $\mathcal{U}_i$ first computes a hash value $m = m_1 \ldots m_n = H(j||M) \in \{0,1\}^n$ and conducts the following steps.

1. Choose $\delta, r \xleftarrow{R} \mathbb{Z}_p^*$ and first compute

$$T_1 = g^\delta \qquad\qquad T_2 = e(h_j, K_2)^\delta \tag{4}$$

as well as

$$\theta_1 = K_1 = (g^\alpha)^{1/(\omega+s_i)} \tag{5}$$
$$\theta_2 = K_2 = g^{s_i} \tag{6}$$
$$\theta_3 = K_3 \cdot F(m)^r = u^{s_i} \cdot F(m)^r \tag{7}$$
$$\theta_4 = g^r \tag{8}$$
$$\theta_5 = h_j^\delta, \tag{9}$$

where $F(m) = v_0 \cdot \prod_{k=1}^n v_k^{m_k}$.

2. Commit to group elements θ_ℓ, for $\ell \in \{1,\ldots,5\}$. For $\ell \in \{1,\ldots,5\}$, choose $r_\ell, s_\ell, t_\ell \xleftarrow{R} \mathbb{Z}_p^*$ and set $\vec{\sigma_\ell} = (1,1,\theta_\ell) \cdot \vec{g_1}^{r_\ell} \cdot \vec{g_2}^{s_\ell} \cdot \vec{g_3}^{t_\ell}$.
3. Give NIWI proofs that committed variables $\theta_1,\ldots,\theta_4$ satisfy

$$e(\theta_1, \Omega \cdot \theta_2) = A \tag{10}$$
$$e(\theta_3, g) = e(u, \theta_2) \cdot e(F(m), \theta_4) \tag{11}$$

Relation (10) is a quadratic pairing product equation (in the Groth-Sahai terminology) over variables θ_1, θ_2. Such a relation requires a proof consisting of 9 group elements that we denote by $\pi_1 = (\vec{\pi}_{1,1}, \vec{\pi}_{1,2}, \vec{\pi}_{1,3})$. Relation (11) is a linear pairing product equation over the variables $\theta_2, \theta_3, \theta_4$. The corresponding proof, that we denote by $\pi_2 = (\pi_{2,1}, \pi_{2,2}, \pi_{2,3}) \in \mathbb{G}^3$, consists of 3 group elements.

5. Give NIZK proofs that committed variables θ_2 and θ_5 satisfy

$$T_2 = e(\theta_2, \theta_5) \tag{12}$$
$$e(h_j, T_1) = e(g, \theta_5) \tag{13}$$

These are two linear pairing product equations over the variables θ_2 and θ_5 and proving them in NIZK requires to introduce an auxiliary variable θ_6. Proving (13) is achieved by proving in NIZK that $e(\theta_6, T_1) = e(g, \theta_5)$ and $\theta_6 = h_j$. The proof for (13) thus comprises an auxiliary commitment $\vec{\sigma_6} = \iota(h_j) \odot \vec{g_1}^{r_6} \odot \vec{g_2}^{s_6} \odot \vec{g_3}^{t_6}$ to $\theta_6 = h_j$ and proofs that relations

$$e(\theta_6, T_1) = e(g, \theta_5) \tag{14}$$
$$e(\theta_6, g) = e(h_j, g) \tag{15}$$

are simultaneously satisfied. These relations are all pairing-product equations. Relation (12) is quadratic and costs 9 group elements to prove. We will call this proofs $\pi_3 = (\vec{\pi}_{3,1}, \vec{\pi}_{3,2}, \vec{\pi}_{3,3})$. Relations (14)-(15) are linear and only require 3 group elements each. The corresponding proofs are denoted by $\pi_4 = (\pi_{4,1}, \pi_{4,2}, \pi_{4,3})$ and $\pi_5 = (\pi_{5,1}, \pi_{5,2}, \pi_{5,3})$.

The signature consists of $\sigma = (T_1, T_2, \vec{\sigma_1}, \ldots, \vec{\sigma_6}, \pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$.

Verify$(j, M, \sigma, \mathsf{gpk}, RL_j)$**:** parse σ as $(T_1, T_2, \vec{\sigma_1}, \ldots, \vec{\sigma_6}, \pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$ and return "`valid`" if and only if all proof are valid and σ passes the revocation test:

1. We abstracted away the construction of proof elements $\pi_1, \pi_2, \pi_3, \pi_4, \pi_5$ for clarity. To explain to proof of anonymity, it will be useful to outline what verification equations look like: namely, $\pi_1, \pi_2, \pi_3, \pi_4, \pi_5$ must satisfy

$$\begin{aligned}
&1)\ F\big(\vec{\sigma_1}, \iota(\Omega)\cdot\vec{\sigma_2}\big) = \iota_T\big(A\big) \odot F\big(\vec{g_1}, \vec{\pi}_{1,1}\big) \odot F\big(\vec{g_2}, \vec{\pi}_{1,2}\big) \odot F\big(\vec{g_3}, \vec{\pi}_{1,3}\big)\\
&2)\ E\big(g, \vec{\sigma_3}\big) = E\big(u, \vec{\sigma_2}\big) \odot E\big(F(m), \vec{\sigma_4}\big)\\
&\qquad\qquad \odot E\big(\pi_{2,1}, \vec{g_1}\big) \odot E\big(\pi_{2,2}, \vec{g_2}\big) \odot E\big(\pi_{2,3}, \vec{g_3}\big)\\
&3)\ F\big(\vec{\sigma_2}, \vec{\sigma_5}\big) = F(\iota(T_2)) \odot F\big(\vec{\pi}_{3,1}, \vec{g_1}\big) \odot F\big(\vec{\pi}_{3,2}, \vec{g_2}\big) \odot F\big(\vec{\pi}_{3,3}, \vec{g_3}\big)\\
&4)\ E\big(T_1, \vec{\sigma_6}\big) = E\big(\iota(g), \vec{\sigma_5}\big) \odot E\big(\pi_{4,1}, \vec{g_1}\big) \odot E\big(\pi_{4,2}, \vec{g_2}\big) \odot E\big(\pi_{4,3}, \vec{g_3}\big)\\
&5)\ E\big(g, \vec{\sigma_6}\big) = E\big(h_j, \iota(g)\big) \odot E\big(\pi_{5,1}, \vec{g_1}\big) \odot E\big(\pi_{5,2}, \vec{g_2}\big) \odot E\big(\pi_{5,3}, \vec{g_3}\big)
\end{aligned}$$

2. The signer must not be revoked at period j: for all $B_{ij} = h_j^{s_i} \in RL_j$,

$$T_2 \neq e(B_{ij}, T_1) \tag{16}$$

As in all VLR-GS schemes, there is an implicit tracing algorithm that can determine which group member created a valid signature using the vector of revocation tokens (and the revocation test (16)) which acts as a tracing key. We observe that, if necessary, the group manager is able to explicitly open the signature in $O(1)$ time by performing a BonehBoyenShacham2004-decryption of $\vec{\sigma_2}$ using the trapdoor information α_1, α_2.

As far as efficiency goes, signatures consist of 46 elements of $\mathbb{G}$ and 1 element of $\mathbb{G}_T$. If we consider an implementation using symmetric pairings with a 256-bit group order and also assume that elements of $\mathbb{G}_T$ have a 1024-bit representation (with symmetric pairings and supersingular curves, such pairing-values can even be compressed to the third of their length as suggested in [34]), we obtain signatures of about 1.56 kB.

3.2 Security

When proving the BU-anonymity property, it seems natural to use a sequence of games starting with the real attack game and ending with a game where T_2 is replaced by a random element of $\mathbb{G}_T$ so as to leave no advantage to the adversary while avoiding to affect the adversary's view provided the Decision Bilinear Diffie-Hellman (DBDH) assumption holds. The problem becomes to simulate (using a fake common reference string) the NIZK proof that (g, T_1, h_j, K_2, T_2) forms a bilinear Diffie-Hellman tuple. Since T_2 is a given element of $\mathbb{G}_T$ in the proof, there is apparently no way to simulate the proof for relation (12).

As a natural workaround to this problem, we use the Decision Tripartite Diffie-Hellman assumption instead of the DBDH assumption in the last transition of the sequence of games.

Theorem 1 (BU-anonymity). *The scheme satisfies the backward unlinkable anonymity assuming that the Decision Linear problem and the Decision Tripartite Diffie-Hellman problem are both hard in $\mathbb{G}$. More precisely, we have*

$$\mathbf{Adv}_{\mathcal{A}}^{\text{bu-anon}}(\lambda) \leq T \cdot N \cdot \left(2 \cdot \mathbf{Adv}_{\mathbb{G}}^{\text{DLIN}}(\lambda) + \mathbf{Adv}_{\mathbb{G}}^{\text{DTDH}}(\lambda)\right) \tag{17}$$

where N is the maximal number of users and T is the number of time periods.

Proof. The proof is a sequence of games organized in such a way that even an unbounded adversary has no advantage in the final game while the first one is the real attack game as captured by definition 3. Throughout the sequence, we call S_i the event that the adversary wins and her advantage is $Adv_i = |\Pr[S_i] - 1/2|$.

Game 1: the challenger $\mathcal{B}$ sets up the scheme by choosing random exponents

$$\omega, \alpha, \alpha_1, \alpha_2, \xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$$

and setting g^ω and $A = e(g,g)^\alpha$. It also sets $u = g^\gamma$ for a randomly chosen $\gamma \xleftarrow{R} \mathbb{Z}_p^*$ and picks $h_1, \ldots, h_T \in \mathbb{G}$ as well as vectors $\overline{v} \in \mathbb{G}^{n+1}$, and defines $\vec{g_1} = (g_1 = g^{\alpha_1}, 1, g)$, $\vec{g_2} = (1, g_2 = g^{\alpha_2}, g)$, $\vec{g_3} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2}$. Using ω, α, it generates users' private keys and answers all queries as in the real game. At the challenge phase, the adversary chooses two unrevoked and uncorrupted users $i_0^\star, i_1^\star$ and is given a challenge signature $\sigma^\star$ on behalf of signer $i_{d^\star}^\star$. Eventually, she outputs a guess $d' \in \{0,1\}$ and her advantage is $Adv_1 = |\Pr[S_1] - 1/2|$, where S_1 denotes the event that $d' = d^\star$.

Game 2: we modify the simulation and let the simulator $\mathcal{B}$ pick two indices $i^\star \in \{1, \ldots, N\}, j^\star \xleftarrow{R} \{1, \ldots, T\}$ at the outset of the simulation. In the challenge phase, $\mathcal{B}$ aborts if $\mathcal{A}$'s chosen pair $(i_0^\star, i_1^\star)$ does not contain $i^\star$ or if $\mathcal{A}$ does not choose to be challenged for period $j^\star$. It also fails

if $i^\star$ is ever queried for corruption or if it is queried for revocation before or during period $j^\star$. Assuming that $\mathcal{B}$ is lucky when drawing $i^\star, j^\star$ (which is the case with probability $(2/N) \cdot (1/T)$ since $i^\star$ and $j^\star$ are independent of $\mathcal{A}$'s view), the introduced failure event does not occur. We can write $Adv_2 = 2 \cdot Adv_1/(NT)$.

Game 3: we introduce a new rule that causes $\mathcal{B}$ to abort. At the challenge step, we have $i^\star \in \{i_0^\star, i_1^\star\}$ unless the failure event of Game 2 occurs. The new rule is the following: when $\mathcal{B}$ flips $d^\star \xleftarrow{R} \{0,1\}$, it aborts if $i_{d^\star}^\star \neq i^\star$. With probability $1/2$, this rule does not apply and we have $Adv_3 = 1/2 \cdot Adv_2$.

Game 4: we modify the setup phase and consider group elements $Z_1 = g^{z_1}$, $Z_2 = g^{z_2}$ that are used to generate the public key gpk and users' private keys. Namely, for $j \in \{1, \ldots, T\} \backslash \{j^\star\}$, $\mathcal{B}$ chooses $\mu_j \xleftarrow{R} \mathbb{Z}_p^*$ and defines $h_j = g^{\mu_j}$ whereas it sets $h_{j^\star} = Z_2$. Also, $\mathcal{B}$ chooses $\nu \xleftarrow{R} \mathbb{Z}_p^*$ and sets $A = e(g, Z_1 \cdot g^\omega)^\nu$ (so that α is implicitly fixed as $\alpha = \nu(z_1 + \omega)$). Private keys of users $i \neq i^\star$ are calculated as $(K_1, K_2, K_3) = \big((Z_1 \cdot g^\omega)^{\nu/(\omega + s_i)}, g^{s_i}, u^{s_i}\big)$, for a random $s_i \xleftarrow{R} \mathbb{Z}_p^*$ and using ω. Since $\mathcal{B}$ knows s_i for each $i \neq i^\star$, it can compute revocation tokens $B_{ij} = h_j^{s_i}$ for users $i \neq i^\star$ in any period.

The group signing key of the expected target user $i^\star$ is set as the triple $(K_1, K_2, K_3) = (g^\nu, Z_1, Z_1^\gamma)$, which implicitly defines $s_{i^\star} = z_1 = \log_g(Z_1)$. We note that, for periods $j \neq j^\star$, the revocation tokens $h_j^{s_{i^\star}}$ are also computable as $Z_2^{\mu_j}$. On the other hand, the token $h_{j^\star}^{s_{i^\star}} = g^{z_1 z_2}$ is not computable from Z_1, Z_2. However, unless the abortion rule of Game 2 occurs, $\mathcal{A}$ does not query it. Although $\mathcal{B}$ does not explicitly use $z_1 = \log_g(Z_1)$ and $z_2 = \log_g(Z_2)$, it still knows all users' private keys and it can use them to answer signing queries according to the specification of the signing algorithm. It comes that $\mathcal{A}$'s view is not altered by these changes and we have $\Pr[S_4] = \Pr[S_3]$.

Game 5: we bring a new change to the setup phase and generate the CRS $(\vec{g_1}, \vec{g_2}, \vec{g_3})$ by setting $\vec{g_3} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2} \odot \iota(g)^{-1}$ instead of $\vec{g_3} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2}$. We note that vectors $\vec{g_1}, \vec{g_3}, \vec{g_3}$ are now linearly independent. Any noticeable change in the adversary's behavior is easily seen[1] to imply a statistical test for the Decision Linear problem so that we can write $|\Pr[S_5] - \Pr[S_4]| = 2 \cdot \mathbf{Adv}^{\mathrm{DLIN}}(\mathcal{B})$.

Game 6: we modify the generation of the challenge signature and use the trapdoor (ξ_1, ξ_2) of the CRS to simulate NIZK proofs. We suppose that $\mathcal{B}$ knows values $(Z_1, Z_2, Z_3) = (g^{z_1}, g^{z_2}, g^{z_3})$ and $\eta = g^{z_1 z_2 z_3}$. Elements Z_1 and Z_2 are used to define the group public key as in Game 4 whereas Z_3 will be used to create the challenge signature on behalf of user $i^\star$ for period $j^\star$. To this end, $\mathcal{B}$ first implicitly defines $\delta = z_3$ by setting

$$T_1 = Z_3 \qquad\qquad T_2 = e(g, \eta).$$

Elements $\theta_1, \ldots, \theta_4$ are committed to as specified by the scheme and π_1, π_2 are calculated accordingly. This time however, $\vec{\sigma_5}$ is calculated as a commitment to $1_{\mathbb{G}}$: namely, $\vec{\sigma_5} = \vec{g_1}^{r_5} \odot \vec{g_2}^{s_5} \odot \vec{g_3}^{t_5}$, where $r_5, s_5, t_5 \xleftarrow{R} \mathbb{Z}_p^*$. Then, $\mathcal{B}$ generates a proof $\pi_3 = (\vec{\pi}_{3,1}, \vec{\pi}_{3,2}, \vec{\pi}_{3,3})$ satisfying

$$F\big(\vec{\sigma_2}, \vec{\sigma_5}\big) = F\big(\iota(g), \iota(\eta)\big) \odot F\big(\vec{\pi}_{3,1}, \vec{g_1}\big) \odot F\big(\vec{\pi}_{3,2}, \vec{g_2}\big) \odot F\big(\vec{\pi}_{3,3}, \vec{g_3}\big). \tag{18}$$

Such an assignment can be obtained as

$$\vec{\pi}_{3,1} = \vec{\sigma_2}^{r_5} \odot \iota(\eta)^{-\xi_1} \qquad \vec{\pi}_{3,2} = \vec{\sigma_2}^{s_5} \odot \iota(\eta)^{-\xi_2} \qquad \vec{\pi}_{3,3} = \iota(\eta) \odot \vec{\sigma_2}^{t_5}.$$

We note that the value $\theta_5 = h_{j^\star}^{\delta} = g^{z_2 z_3}$ is not used by $\mathcal{B}$. To simulate the proof π_3 that $T_2 = e(\theta_2, \theta_5)$ without knowing θ_5, the simulator takes advantage of the fact that $T_2 = e(g, \eta)$

[1] Indeed, $\Pr[\mathcal{B}(g_1, g_2, g_1^{\xi_1}, g_2^{\xi_2}, g^{\xi_1+\xi_2}) = 1]$ and $\Pr[\mathcal{B}(g_1, g_2, g_1^{\xi_1}, g_2^{\xi_2}, g^{\xi_1+\xi_2-1}) = 1]$ are both within distance $\mathbf{Adv}^{\mathrm{DLIN}}(\mathcal{B})$ from $\Pr[\mathcal{B}(g_1, g_2, g_1^{\xi_1}, g_2^{\xi_2}, g^z) = 1]$, where z is random.

for known $g, \eta \in \mathbb{G}$ (and simulating such a proof would not have been possible if T_2 had been a given element of $\mathbb{G}_T$). To simulate proofs $\pi_4 = (\pi_{4,1}, \pi_{4,2}, \pi_{4,3})$, $\pi_5 = (\pi_{5,1}, \pi_{5,2}, \pi_{5,3})$ that relations (14)-(15) are both satisfied, $\mathcal{B}$ generates π_4 as if it were a real proof using the variable assignment $\theta_5 = \theta_6 = 1_{\mathbb{G}}$ that obviously satisfies $e(\theta_6, T_1) = e(g, \theta_5)$ (and $\vec{\sigma_6} = \vec{g_1}^{r_6} \odot \vec{g_2}^{s_6} \odot \vec{g_3}^{t_6}$ is thus computed as a commitment to $1_{\mathbb{G}}$). As for π_5, the assignment

$$\pi_{5,1} = g^{r_6} \cdot h_j^{-\xi_1} \qquad \pi_{5,2} = g^{s_6} \cdot h_j^{-\xi_2} \qquad \pi_{5,3} = g^{t_6} \cdot h_j.$$

is easily seen to satisfy the last verification equation

$$E\big(g, \vec{\sigma_6}\big) = E\big(h_j, \iota(g)\big) \odot E\big(\pi_{5,1}, \vec{g_1}\big) \odot E\big(\pi_{5,2}, \vec{g_2}\big) \odot E\big(\pi_{5,3}, \vec{g_3}\big)$$

since $\vec{g_3} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2} \odot \iota(g)^{-1}$. Simulated proofs π_4, π_5 are then randomized as explained in [24] to be uniform in the space of valid proofs and achieve perfect witness indistinguishability. Simulated proofs are perfectly indistinguishable from real proofs and $\Pr[S_6] = \Pr[S_5]$.

Game 7: is identical to Game 6 but we replace η (that was equal to $g^{z_1 z_2 z_3}$ in Game 6) by a random group element. It is clear that, under the DTDH assumption, this change does not significantly alter $\mathcal{A}$'s view. We thus have $|\Pr[S_7] - \Pr[S_6]| \leq \mathbf{Adv}^{\mathrm{DTDH}}_{\mathbb{G},\mathcal{B}}(\lambda)$.

In Game 7, it is easy to see that $\Pr[S_7] = 1/2$. Elements T_1 and T_2 are indeed completely independent of $s_{i^\star} = z_1$ (and thus of $i^\star$). Moreover, in the WI setting, all commitments $\vec{\sigma_1}, \ldots, \vec{\sigma_5}$ are perfectly hiding and proofs $\pi_1, \ldots, \pi_5$ reveal no information on underlying witnesses.

When gathering probabilities, we obtain the upper bound (17) on $\mathcal{A}$'s advantage in Game 1. □

Theorem 2 (Traceability). *The scheme satisfies the full non-traceability assuming that the N-Hidden Strong Diffie-Hellman problem is hard in $\mathbb{G}$. More precisely, we have*

$$\mathbf{Adv}^{\mathrm{trace}}_{\mathcal{A}}(\lambda) \leq 4 \cdot n \cdot N \cdot q_s \cdot \left(1 - \frac{(N-1)}{p}\right)^{-1} \cdot \Big(\mathbf{Adv}^{N\text{-HSDH}}(\lambda) + \mathbf{Adv}^{\mathrm{CR}}(n)\Big) \quad (19)$$

where N is maximum of the number of the adversary signature queries and the maximal number of users and T is the number of time periods.

Proof. The proof is very similar to the proof of full traceability in the Boyen-Waters [11] group signature. One difference is that [11] reduces the full traceability property of their scheme to the unforgeability of a 2-level hierarchical signature [28]. To prove this result, Boyen and Waters restricted the message space (where the element s_i, that uniquely identifies the group member is the group signature, must be chosen) to a relatively small interval at the first level.

In our proof of anonymity, we need elements s_i to be uniformly chosen in $\mathbb{Z}_p^*$. Therefore, we cannot directly link the security of our scheme to that of the 2-level hierarchical signature of [11] and a direct proof is needed (but it is simply obtained using the techniques from [11]). Namely, two kinds of forgeries must be considered as in [11]:

- **Type I forgeries** are those for which the implicit tracing algorithm fails to identify the signer using the vector of revocation tokens for the relevant period $j^\star$.
- **Type II forgeries** are those for which the implicit tracing algorithm incriminates a user outside the coalition and that was not requested to sign the message $M^\star$ during period $j^\star$.

The two kinds of adversaries are handled separately in lemmas 1 and 2.

To conclude the proof, we consider an algorithm $\mathcal{B}$ that guesses the kind of forgery that $\mathcal{A}$

will come up with. Then, $\mathcal{B}$ runs the appropriate HSDH solver among those described in previous lemmas. If the guess is correct, $\mathcal{B}$ solves the HSDH problem with the success probability given in the lemmas. Since this guess is correct with probability $1/2$, we obtain the claimed security bound. □

Lemma 1. *If N is the maximal number of users, any Type I forger $\mathcal{A}$ has no advantage than* $\mathbf{Adv}_{\mathcal{A}}^{\text{Type-I}}(\lambda) \leq \mathbf{Adv}^{N\text{-HSDH}}(\lambda)$.

Proof. The proof is close to the one of lemma A.1 in [11]. The simulator $\mathcal{B}$ is given a N-HSDH instance consisting of elements $(g, \Omega = g^{\omega}, u)$ and triples $\{(A_i, B_i, C_i) = (g^{1/(\omega+s_i)}, g^{s_i}, u^{s_i})\}_{i=1,\ldots,N}$.

The simulator picks $\alpha, \beta_0, \ldots, \beta_n \xleftarrow{R} \mathbb{Z}_p^*$ and sets $v_i = g^{\beta_i}$, for $i = 0, \ldots, n$. Vectors $\vec{g_1}, \vec{g_2}, \vec{g_3}$ are chosen as $\vec{g_1} = (g_1 = g^{\alpha_1}, 1, g)$, $\vec{g_2} = (1, g_2 = g^{\alpha_2}, g)$ and $\vec{g_3} = \vec{g_1}^{\xi_1} \odot \vec{g_2}^{\xi_2}$, for randomly chosen $\alpha_1, \alpha_2, \xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$, in such a way that the CRS $\mathbf{g} = (\vec{g_1}, \vec{g_2}, \vec{g_3})$ provides perfectly sound proofs for which $\mathcal{B}$ retains the extraction trapdoor $(\alpha_1 = \log_g(g_1), \alpha_2 = \log_g(g_2))$. Finally, $\mathcal{B}$ generates $(h_1, \ldots, h_T) \in \mathbb{G}^T$ as $h_j = g^{\zeta_j}$, for $j = 1, \ldots, T$, with $\zeta_1, \ldots, \zeta_T \xleftarrow{R} \mathbb{Z}_p^*$. Then, $\mathcal{B}$ starts interacting with the Type I adversary $\mathcal{A}$ who is given the group public key $\mathsf{gpk} := (g, A = e(g,g)^{\alpha}, h_1, \ldots, h_T, \Omega, u, \overline{v}, \mathbf{g})$ and the vector of revocation tokens grt, which $\mathcal{B}$ generates as $\mathsf{grt}[i][j] = h_j^{s_i} = B_i^{\zeta_j}$. The simulation proceeds as follows:

- when $\mathcal{A}$ decides to corrupt user $i \in \{1, \ldots, N\}$, $\mathcal{B}$ returns the HDSH triple (A_i, B_i, C_i).
- when $\mathcal{A}$ queries a signature from user $i \in \{1, \ldots, N\}$ for a message M, $\mathcal{B}$ uses the private key $(K_1, K_2, K_3) = (A_i, B_i, C_i)$, to generate the signature by following the specification of the signing algorithm.

When $\mathcal{A}$ outputs her forgery $(M^{\star}, j^{\star}, \sigma^{\star})$, $\mathcal{B}$ uses elements α_1, α_2 to decrypt $\vec{\sigma_i}^{\star}$, for indices $i \in \{1, \ldots, 5\}$, and obtain $\theta_1^{\star} = (g^{\alpha})^{1/(\omega+s^{\star})}$, $\theta_2^{\star} = g^{s^{\star}}$ as well as $\theta_3^{\star} = u^{s^{\star}} \cdot (v_0 \cdot \prod_{k=1}^{n} v_k^{m_k})^r$ and $\theta_4^{\star} = g^r$. From these values, $\mathcal{B}$ can extract $u^{s^{\star}}$ since it knows the discrete logarithm $\log_g(v_0 \cdot \prod_{k=1}^{n} v_k^{m_k}) = \beta_0 + \sum_{k=1}^{n} m_k \beta_k$, where $m_1 \ldots m_n = H(j^{\star} || M^{\star}) \in \{0,1\}^n$. Since $\sigma^{\star}$ is a Type I forgery, the implicit tracing algorithm must fail to identify one of the group members $\{1, \ldots, N\}$. The perfect soundness of the proof system implies that $s^{\star} \notin \{s_1, \ldots, s_N\}$ and $(\theta_1^{\star 1/\alpha}, \theta_2^{\star}, u^{s^{\star}})$ is necessarily an acceptable solution. □

Lemma 2. *The scheme is secure against Type II forgeries under the $(N-1)$-HSDH assumption. The advantage of any Type II adversary $\mathcal{A}$ is at most*

$$\mathbf{Adv}_{\mathcal{A}}^{\text{Type-II}}(\lambda, n) \leq 2 \cdot n \cdot N \cdot q_s \cdot \left(1 - \frac{(N-1)}{p}\right)^{-1} \cdot \left(\mathbf{Adv}^{(N-1)\text{-HSDH}}(\lambda) + \mathbf{Adv}^{\text{CR}}(n)\right)$$

where N and q_s stand for the number of users and the number of signing queries, respectively, and the last term accounts for the probability of breaking the collision-resistance of H.

Proof. The proof is based on lemma A.2 in [11]. Namely, the simulator $\mathcal{B}$ receives a $(N-1)$-HSDH input comprising $(g, \Omega = g^{\omega}, u)$ and a set of triples

$$\{(A_i, B_i, C_i) = (g^{1/(\omega+s_i)}, g^{s_i}, u^{s_i})\}_{i=1,\ldots,N-1}.$$

To prepare the public key gpk, the simulator $\mathcal{B}$ picks a random index $\nu \xleftarrow{R} \{0, \ldots, n\}$, as well as $\rho_0, \ldots, \rho_n \xleftarrow{R} \mathbb{Z}_p^*$ and integers $\beta_0, \ldots, \beta_n \xleftarrow{R} \{0, \ldots, 2q_s - 1\}$. It sets $v_0 = u^{\beta_0 - 2\nu q_s} \cdot g^{\rho_0}$, $v_i = u^{\beta_i} \cdot g^{\rho_i}$ for $i = 1, \ldots, n$. It also defines $h_1, \ldots, h_T$ by setting $h_j = g^{\zeta_j}$, with $\zeta_j \xleftarrow{R} \mathbb{Z}_p^*$, for $j = 1, \ldots, T$. It finally chooses vectors $\mathbf{g}$ as specified by the setup algorithm to obtain perfectly sound proofs.

Before starting its interaction with the Type II forger $\mathcal{A}$, $\mathcal{B}$ initializes a counters $ctr \leftarrow 0$

and chooses an index $i^\star \stackrel{R}{\leftarrow} \{1,\ldots,N\}$ as a guess for the honest user on behalf of which $\mathcal{A}$ will attempt to generate a forgery. The simulation proceeds by handling $\mathcal{A}$'s queries in the following way.

Queries: at the first time that user $i \in \{1,\ldots,N\}$ is involved in a signing query or a corruption query, $\mathcal{B}$ does the following:

- if the query is a corruption query, $\mathcal{B}$ halts and declares failure if $i = i^\star$ as it necessarily guessed the wrong user $i^\star$. Otherwise, it increments ctr and returns the triple $(A_{ctr}, B_{ctr}, C_{ctr})$ as a private key for user (K_1, K_2, K_3).
- if the query is a signing query for period $j \in \{1,\ldots,T\}$,
  - if $i \neq i^\star$ $\mathcal{B}$ increments ctr and answers the query by running the signing algorithm using the private key $(K_1, K_2, K_3) = (A_{ctr}, B_{ctr}, C_{ctr})$.
  - if $i = i^\star$, $\mathcal{B}$ chooses $t^\star \stackrel{R}{\leftarrow} \mathbb{Z}_p^*$ at random and implicitly defines a triple $(K_1^\star, K_2^\star, K_3^\star) = (g^{1/t^\star}, g^{t^\star} \cdot \Omega^{-1}, *)$, where $*$ is a placeholder for an unknown group element (note that this implicitly defines $s^* = t^\star - \omega$). Then, $\mathcal{B}$ computes $m_1 \ldots m_n = H(j||M) \in \{0,1\}^n$. At this stage, it is convenient to write $F(m_1 \ldots m_n) = v_0 \cdot \prod_{k=1}^n v_k^{m_k}$ as $F(m_1 \ldots m_n) = u^J \cdot g^K$ where $J = \beta_0 - 2\nu q_s + \sum_{j=1}^n \beta_j m_j$, $K = \rho_0 + \sum_{j=1}^n \rho_j m_j$. If $J = 0$, $\mathcal{B}$ aborts. Otherwise, it can pick $r \stackrel{R}{\leftarrow} \mathbb{Z}_p^*$ and compute a pair

$$\left(\theta_3 = u^{t^\star} \cdot F(m_1 \ldots m_n)^r \cdot \Omega^{\frac{K}{J}},\ \theta_4 = g^r \cdot \Omega^{\frac{1}{J}}\right),$$

    which can be re-written as $(\theta_4 = u^{t^\star - \omega} \cdot F(m_1 \ldots m_n)^{\tilde{r}}, \theta_5 = g^{\tilde{r}})$ if we define $\tilde{r} = r + \omega/J(\mathsf{m})$. This pair then allows generating a suitably anonymized signature. In particular, since $\mathcal{B}$ knows $\theta_2 = K_2^\star = g^{t^\star} \cdot \Omega^{-1}$, it is able to compute $T_2 = e(h_j, K_2^\star)^\delta$ and $T_1 = g^\delta$ for a random $\delta \stackrel{R}{\leftarrow} \mathbb{Z}_p^*$.

When subsequent queries involve the same user i, $\mathcal{B}$ responds as follows (we assume that corruption queries are distinct):

– For corruption queries on users $i \in \{1,\ldots,N\}$ that were previously involved in signing queries, $\mathcal{B}$ aborts if $i = i^\star$. Otherwise, it knows the private key (K_1, K_2, K_3) (that was used to answer signing queries) and hands it to $\mathcal{A}$.
– For signing queries, $\mathcal{B}$ uses the same values as in the first query involving the user $i \in \{1,\ldots,N\}$. If $i \neq i^\star$, $\mathcal{B}$ uses the same triple $(A_{ctr}, B_{ctr}, C_{ctr})$. In the case $i = i^\star$, $\mathcal{B}$ re-uses the pair $(K_1^\star, K_2^\star) = (g^{1/t^\star}, g^{t^\star} \cdot \Omega^{-1})$ and proceeds as in the first query involving $i^\star$ (but uses a fresh random exponent r).

Forgery: the game ends with the adversary outputting message $M^\star$ together with a type II forgery $\sigma^\star = (T_1^\star, T_2^\star, \vec{\sigma_1}^\star, \ldots, \vec{\sigma_6}^\star, \pi_1^\star, \ldots, \pi_5^\star)$ for some period $j^\star \in \{1,\ldots,T\}$. By assumption, the implicit tracing algorithm must point to some user who did not sign $M^\star$ at period $j^\star$. Then, $\mathcal{B}$ halts and declares failure if $\sigma^\star$ does not trace to user $i^\star$. Since the chosen index $i^\star$ was independent of $\mathcal{A}$'s view, with probability $1/N$, $\mathcal{B}$'s guess turns out to be correct. Then, the perfect soundness of the proof system implies that $\vec{\sigma_2}^\star$ is a BonehBoyenShacham2004 encryption of $K_2^\star$. Then, $\mathcal{B}$ computes $\mathsf{m}^\star = m_1 \ldots m_n = H(j^\star||M^\star)$. If user $i^\star$ signed a message M at period j such that $(j, M) \neq (j^\star, M^\star)$ but $H(j||M) = H(j^\star||M^\star)$, $\mathcal{A}$ was necessarily able to generate a collision on H. Otherwise, the perfect soundness of the proof system implies that $\vec{\sigma_3}^\star$ and $\vec{\sigma_4}^\star$ decrypt into

$$\theta_3^\star = u^{t^\star - \omega} F(\mathsf{m}^\star)^r \qquad\qquad \theta_4^\star = g^r$$

for some $r \in \mathbb{Z}_p^*$ and where $F(\mathsf{m}^\star) = v_0 \cdot \prod_{k=1}^{n} v_k^{m_k} = u^{J^\star} \cdot g^{K^\star}$ and $s_\star = t_{i^\star} - \omega$. Then, $\mathcal{B}$ aborts if $J(\mathsf{m}^\star) = \beta_0 + \sum_{j=1}^{n} \beta_j m_j - 2\nu q_s \neq 0$. Otherwise, $\mathcal{B}$ can compute u^{s^*} and thereby obtains a full tuple $\left(g^{1/(\omega+s^\star)},\ g^{s^\star},\ u^{s^\star}\right)$ where $s^\star = t^\star - \omega$ differs from $s_1, \ldots, s_{N-1}$ with probability at least $1 - (N-1)/p$ (since the value $t^\star$ was chosen at random).

$\mathcal{B}$'s probability not to abort throughout the simulation can assessed as in [36, 11]. More precisely, one can show that $J \neq 0$ in all signing queries with probability greater than $1/2$. Conditionally on the event that $\mathcal{B}$ does not abort before the forgery stage, the probability to have $J^\star = 0$ is then shown to be at least $1/(2nq_s)$ (see [36, 11] for details). □

3.3 A Variant with Shorter Group Public Keys

As described in this section, the scheme suffers from a group public key of size $O(T)$, which makes it impractical when the number of time periods is very large. In the random oracle model $h_1, \ldots, h_T$ could be derived from a random oracle. However, avoiding the dependency on T in the group public key size is also possible without resorting to random oracles. This can be achieved using the techniques introduced in [6] in the context of identity-based encryption.

The vector $(h_1, \ldots, h_T)$ is replaced by a triple $(h, h_0, h_1) \in \mathbb{G}^3$ and the revocation token of user i at period $j \in \{1, \ldots, T\}$ is defined to be the pair $(B_{ij1}, B_{ij2}) = (h^{s_i} \cdot F(j)^\rho, g^\rho)$, where $\rho \xleftarrow{R} \mathbb{Z}_p^*$ and $F(j) = h_0 \cdot h_1^j$ is the selectively-secure identity-hashing function of Boneh and Boyen [6]. Since the revocation token (B_{ij1}, B_{ij2}) satisfies the relation $e(B_{ij1}, g) = e(h, g^{s_i}) \cdot e(F(j), B_{ij2})$, we have $e(B_{ij1}, g^\delta) = e(h, g^{s_i})^\delta \cdot e(F(j)^\delta, B_{ij2})$ for any $\delta \in \mathbb{Z}_p^*$.

Therefore, in each signature σ, the pair (T_1, T_2) is superseded by a triple $(T_1, T_2, T_3) = (g^\delta, F(j)^\delta, e(h, K_2)^\delta)$ (so that the verifier needs the check that $e(T_1, F(j)) = e(g, T_2)$) whereas $\vec{\sigma_5}$ becomes a commitment to $\theta_5 = h^\delta$ and the NIZK proof for relation (13) is replaced by a proof that $e(h, T_1) = e(g, \theta_5)$. At step 2 of the verification algorithm, the revocation test then consists in testing whether $e(T_1, B_{ij1}) = T_3 \cdot e(T_2, B_{ij2})$ for revocation tokens $\{(B_{ij1}, B_{ij2})\}_{i \in RL_j}$. Using the technique of [6] to generate tokens for periods $j \in \{1, \ldots, T\} \backslash \{j^\star\}$, it can be checked that everything goes through in the proof of anonymity.

4 Conclusion

We described a simple way to provide Boyen-Waters group signatures with an efficient verifier local revocation mechanism with backward unlinkability.

The scheme can be easily extended so as to provide exculpability (and prevent the group manager from signing on behalf of users) using a dynamic joining protocol such as the one of [29]. It would be interesting to turn the scheme into a traceable signature [26] supporting fine-grained (*i.e.* per period) user tracing while leaving users the ability to claim their signatures.

References

1. G. Ateniese, J. Camenisch, S. Hohenberger, and B. de Medeiros, *Practical Group Signatures without Random Oracles.*, Tech. Report 2005/385, IACR eprint, 2005.
2. G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik, *A Practical and Provably Secure Coalition-Resistant Group Signature Scheme.*, Advances in Cryptology - CRYPTO 2000 (M. Bellare, ed.), Lect. Notes Comput. Sci., vol. 1880, Springer, 2000, pp. 255–270.
3. M. Bellare, D. Micciancio, and B. Warinschi, *Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions.*, Advances in Cryptology - EUROCRYPT 2003 (E. Biham, ed.), Lect. Notes Comput. Sci., vol. 2656, Springer, 2003, pp. 614–629.
4. M. Bellare and P. Rogaway, *Random Oracles are Practical: A Paradigm for Designing Efficient Protocols.*, Proceedings of the First ACM Conference on Computer and Communications Security (D. Denning, R. Pyle, R. Ganesan, R. Sandhu, and V. Ashby, eds.), ACM Press, 1993, pp. 62–73.

5. M. Bellare, H. Shi, and C. Zhang, *Foundations of Group Signatures: The Case of Dynamic Groups.*, Topics in Cryptology - CT-RSA 2005 (A. J. Menezes, ed.), Lect. Notes Comput. Sci., vol. 3376, Springer, 2005, pp. 136–153.
6. D. Boneh and X. Boyen, *Short Signatures Without Random Oracles.*, in Cachin and Camenisch [12], pp. 56–73.
7. D. Boneh, X. Boyen, and H. Shacham, *Short Group Signatures.*, in Franklin [19], pp. 41–55.
8. D. Boneh and M. Franklin, *Identity-Based Encryption from the Weil Pairing.*, SIAM J. Comput. **32** (2003), no. 3, 586–615.
9. D. Boneh and H. Shacham, *Group Signatures with Verifier-Local Revocation.*, Proceedings of the 11th ACM Conference on Computer and Communications Security (V. Atluri, B. Pfitzmann, and P. McDaniel, eds.), ACM Press, 2004, pp. 168–177.
10. X. Boyen and B. Waters, *Compact Group Signatures Without Random Oracles.*, Advances in Cryptology - EUROCRYPT 2006 (S. Vaudenay, ed.), Lect. Notes Comput. Sci., vol. 4004, Springer, 2006, pp. 427–444.
11. ———, *Full-Domain Subgroup Hiding and Constant-Size Group Signatures.*, 10th International Workshop on Practice and Theory in Public Key Cryptography, PKC 2007 (T. Okamoto and X. Wang, eds.), Lect. Notes Comput. Sci., vol. 4450, Springer, 2007, pp. 1–15.
12. C. Cachin and J. Camenisch (eds.), *Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings*, Lect. Notes Comput. Sci., vol. 3027, Springer, 2004.
13. R. Canetti, O. Goldreich, and S. Halevi, *The Random Oracle Methodology, Revisited.*, J. Assoc. Comput. Mach. **51** (2004), no. 4, 557–594.
14. D. Chaum, *An Efficient Protocol for Anonymously Providing Assurance of the Container of the Private Key.*, Submission to the Trusted Computing Group, 2003.
15. D. Chaum and E. van Heyst, *Group Signatures.*, Advances in Cryptology - EUROCRYPT'91 (D. W. Davies, ed.), Lect. Notes Comput. Sci., vol. 547, Springer, 1991, pp. 257–265.
16. S. G. Choi, K. Park, and M. Yung, *Short Traceable Signatures Based on Bilinear Pairings.*, in Yoshiura et al. [37], pp. 88–103.
17. C. Delerablée and D. Pointcheval, *Dynamic Fully Anonymous Short Group Signatures.*, Progress in Cryptology - VIETCRYPT 2006 (P. Q. Nguyen, ed.), Lect. Notes Comput. Sci., vol. 4341, Springer, 2006, pp. 193–210.
18. A. Fiat and A. Shamir, *How to Prove Yourself: Practical Solutions to Identification and Signature Problems.*, Advances in Cryptology - CRYPTO'86 (A. M. Odlyzko, ed.), Lect. Notes Comput. Sci., vol. 263, Springer, 1987, pp. 186–194.
19. M. K. Franklin (ed.), *Advances in Cryptology - CRYPTO 2004, 24th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings*, Lect. Notes Comput. Sci., vol. 3152, Springer, 2004.
20. J. Furukawa and H. Imai, *An Efficient Group Signature Scheme from Bilinear Maps.*, Information Security and Privacy: 10th Australasian Conference, ACISP 2005 (C. Boyd and J. M. G. Nieto, eds.), Lect. Notes Comput. Sci., vol. 3574, Springer, 2005, pp. 455–467.
21. S. Goldwasser and Y. Tauman Kalai, *On the (In)security of the Fiat-Shamir Paradigm.*, Proceedings of the 44th IEEE Symposium on Foundations of Computer Science (FOCS 2003) (M. Sudan, ed.), IEEE Computer Society, 2003, pp. 102–113.
22. J. Groth, *Simulation-Sound NIZK Proofs for a Practical Language and Constant Size Group Signatures.*, Advances in Cryptology - ASIACRYPT 2006 (X. Lai and K. Chen, eds.), Lect. Notes Comput. Sci., vol. 4284, Springer, 2006, pp. 444–459.
23. ———, *Fully Anonymous Group Signatures Without Random Oracles.*, Advances in Cryptology - ASIACRYPT 2007 (K. Kurosawa, ed.), Lect. Notes Comput. Sci., vol. 4833, Springer, 2007, pp. 164–180.
24. J. Groth and A. Sahai, *Efficient Non-interactive Proof Systems for Bilinear Groups.*, Advances in Cryptology - EUROCRYPT 2008 (N. P. Smart, ed.), Lect. Notes Comput. Sci., vol. 4965, Springer, 2008, pp. 415–432.
25. D. Hofheinz and E. Kiltz, *Programmable Hash Functions and Their Applications.*, Advances in Cryptology - CRYPTO 2008 (D. Wagner, ed.), Lect. Notes Comput. Sci., vol. 5157, Springer, 2008, pp. 21–38.
26. A. Kiayias, Y. Tsiounis, and M. Yung, *Traceable Signatures.*, in Cachin and Camenisch [12], pp. 571–589.
27. A. Kiayias and M. Yung, *Group Signatures: Provable Security, Efficient Constructions and Anonymity from Trapdoor-Holders.*, Tech. Report 2003/076, IACR eprint, 2003.
28. E. Kiltz, A. Mityagin, S. Panjwani, and B. Raghavan, *Append-Only Signatures.*, Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005 (L. Caires and L. Monteiro, eds.), Lect. Notes Comput. Sci., vol. 3580, Springer, 2005, pp. 434–445.
29. B. Libert and M. Yung, *Efficient Traceable Signatures in the Standard Model.*, Pairing-Based Cryptography - Pairing 2009, Third International Conference (S. D. Galbraith and K. G. Paterson, eds.), Lect. Notes Comput. Sci., vol. 5209, Springer, 2008, pp. 187–205.
30. T. Nakanishi and N. Funabiki, *Verifier-Local Revocation Group Signature Schemes with Backward Unlinkability from Bilinear Maps.* , Advances in Cryptology - ASIACRYPT 2005 (B. Roy, ed.), Lect. Notes Comput. Sci., vol. 3788, Springer, 2005, pp. 533–548.

31. ———, *A Short Verifier-Local Revocation Group Signature Scheme with Backward Unlinkability.*, in Yoshiura et al. [37], pp. 17–32.
32. M. Naor, *On Cryptographic Assumptions and Challenges.*, Advances in Cryptology - CRYPTO 2003 (D. Boneh, ed.), Lect. Notes Comput. Sci., vol. 2729, Springer, 2003, pp. 96–109.
33. L. Nguyen and R. Safavi-Naini, *Efficient and Provably Secure Trapdoor-Free Group Signature Schemes from Bilinear Pairings.*, Advances in Cryptology - ASIACRYPT 2004 (P. J. Lee, ed.), Lect. Notes Comput. Sci., vol. 3329, Springer, 2004, pp. 372–386.
34. M. Scott and P. S. L. M. Barreto, *Compressed Pairings.*, in Franklin [19], pp. 140–156.
35. D. X. Song, *Practical Forward Secure Group Signature Schemes.*, Proceedings of the 8th ACM Conference on Computer and Communications Security (M. Reiter and P. Samarati, eds.), ACM Press, 2001, pp. 225–234.
36. B. Waters, *Efficient Identity-Based Encryption Without Random Oracles.*, Advances in Cryptology - EUROCRYPT 2005 (R. Cramer, ed.), Lect. Notes Comput. Sci., vol. 3494, Springer, 2005, pp. 114–127.
37. H. Yoshiura, K. Sakurai, K. Rannenberg, Y. Murayama, and S.-I. Kawamura (eds.), *Advances in Information and Computer Security, First International Workshop on Security, IWSEC 2006, Kyoto, Japan, October 23-24, 2006, Proceedings*, Lect. Notes Comput. Sci., vol. 4266, Springer, 2006.
38. S. Zhou and D. Lin, *A Shorter Group Signature with Verifier-Location Revocation and Backward Unlinkability.*, Tech. Report 2006/100, IACR eprint, 2006.
39. ———, *Shorter Verifier-Local Revocation Group Signatures from Bilinear Maps.*, Cryptology and Network Security, 5th International Conference, CANS 2006 (D. Pointcheval, Y. Mu, and K. Chen, eds.), Lect. Notes Comput. Sci., vol. 4301, Springer, 2006, pp. 126–143.