Group Signatures with Almost-for-free Revocation

Benoît Libert[1] ⋆, Thomas Peters[1] ⋆⋆, and Moti Yung[2]

[1]Université catholique de Louvain, ICTEAM Institute (Belgium)
[2] Google Inc. and Columbia University (USA)

Abstract. Group signatures are a central cryptographic primitive where users can anonymously and accountably sign messages in the name of a group they belong to. Several efficient constructions with security proofs in the standard model (*i.e.*, without the random oracle idealization) appeared in the recent years. However, like standard PKIs, group signatures need an efficient revocation system to be practical. Despite years of research, membership revocation remains a non-trivial problem: many existing solutions do not scale well due to either high overhead or constraining operational requirements (like the need for all users to update their keys after each revocation). Only recently, Libert, Peters and Yung (Eurocrypt'12) suggested a new scalable revocation method, based on the Naor-Naor-Lotspiech (NNL) broadcast encryption framework, that interacts nicely with techniques for building group signatures in the standard model. While promising, their mechanism introduces important storage requirements at group members. Namely, membership certificates, which used to have constant size in existing standard model constructions, now have polylog size in the maximal cardinality of the group (NNL, after all, is a tree-based technique and such dependency is naturally expected). In this paper we show how to obtain private keys of *constant* size. To this end, we introduce a new technique to leverage the NNL subset cover framework in the context of group signatures but, perhaps surprisingly, without logarithmic relationship between the size of private keys and the group cardinality. Namely, we provide a way for users to efficiently prove their membership of one of the generic subsets in the NNL subset cover framework. This technique makes our revocable group signatures competitive with ordinary group signatures (*i.e.*, without revocation) in the standard model. Moreover, unrevoked members (as in PKIs) still do not need to update their keys at each revocation.

Keywords. Group signatures, revocation, standard model, efficiency, short private keys.

1 Introduction

Group signatures, as suggested by Chaum and van Heyst [29], allow members of a group managed by some authority to sign messages in the name of the group while hiding their identity. At the same time, a tracing authority has the power of identifying the signer if necessary. A crucial problem is the revocation of the anonymous signing capability of users when they are banned from or intentionally leave the group.

1.1 Related Work

Ordinary group signatures. The first efficient and provably coalition-resistant group signature dates back to the work of Ateniese, Camenisch, Joye and Tsudik [6]. By the time their scheme appeared, the security of the primitive was not appropriately formalized yet. Suitable security definitions remained lacking until the work of Bellare, Micciancio and Warinschi [8] (BMW) who captured all the requirements of group signatures in three properties. In (a variant of) this model,

⋆ This author acknowledges the Belgian Fund for Scientific Research (F.R.S.-F.N.R.S.) for his "Collaborateur scientifique" fellowship.
⋆⋆ Supported by the IUAP B-Crypt Project and the Walloon Region Camus Project.

Boneh, Boyen and Shacham [14] obtained very short signatures using the random oracle methodology [9].

The BMW model assumes static groups where no new member can be introduced after the setup phase. The setting of dynamically changing groups was analyzed later on by Bellare-Shi-Zhang [10] and, independently, by Kiayias and Yung [40]. In the models of [10, 40], constructions featuring relatively short signatures were proposed in [54, 30]. A construction in the standard model was also suggested by Ateniese *et al.* [5] under interactive assumptions. At the same time, Boyen and Waters gave a different solution [18] without random oracles using more standard assumptions. By improving upon their own scheme, they managed [19] to obtain signatures of constant size. Their constructions [18, 19] were both presented in the BMW model [8] and provide anonymity in the absence of signature opening oracle. In the dynamic model [10], Groth [34] showed a system in the standard model with $O(1)$-size signatures but, due to very large hidden constants, his scheme was mostly a feasibility result. Later on, Groth came up with an efficient realization [35] (and signatures of about 50 group elements) with the strongest anonymity level.

Revocation. As in ordinary PKIs, where certificate revocation is a critical issue, membership revocation is a complex problem that has been extensively studied [20, 7, 26, 17] in the last decade. Generating a new group public key and distributing new signing keys to unrevoked members is a simple solution. In large groups, it is impractical to update the public key and provide members with new keys after they joined the group. Bresson and Stern suggested a different approach [20] consisting of having the signer prove that his membership certificate does not belong to a list of revoked certificates. Unfortunately, the length of signatures grows with the number of revoked members. In forward-secure group signatures, Song [56] chose a different way to handle revocation but verification takes linear time in the number of excluded users.

Camenisch and Lysyanskaya [26] proposed an elegant method using accumulators[1] [11]. Their technique, also used in [59, 24], allows revoking members while keeping $O(1)$ costs for signing and verifying. The downside of this approach is its history-dependence: it requires users to follow the dynamic evolution of the group and keep track of all changes: each revocation incurs a modification of the accumulator value, so that unrevoked users have to upgrade their membership certificate before signing new messages. In the worst case, this may require up to $O(r)$ exponentiations, if r is the number of revoked users.

Another drawback of accumulator-based approaches is their limited applicability in the standard model. Indeed, for compatibility reasons with the central tool of Groth-Sahai proofs, pairing-based accumulators are the only suitable candidates. However, in known pairing-based accumulators [53, 24], public keys have linear size in the maximal number of accumulations, which would result in linear-size group public keys in immediate implementations. To address this concern in delegatable anonymous credentials, Acar and Nguyen [4] chose to sacrifice the constant size of proofs of non-membership but, in group signatures, this would prevent signatures from having constant size. Boneh, Boyen and Shacham [14] managed to avoid linear dependencies in a revocation mechanism along the lines of [26]. Unfortunately, their technique does not seem to readily interact[2] with Groth-

[1] An accumulator is a kind of "hash" function mapping a set of values to a short, constant-size string while allowing to efficiently prove that a specific value was accumulated.

[2] In [14], signing keys consist of pairs $(g^{1/(\omega+s)}, s) \in \mathbb{G} \times \mathbb{Z}_p$, where $\omega \in \mathbb{Z}_p$ is the secret key of the group manager, and the revocation method relies on the availability of the exponent $s \in \mathbb{Z}_p$. In the standard model, the Groth-Sahai techniques would require to turn the membership certificates into triples $(g^{1/(\omega+s)}, g^s, u^s)$, for some $u \in \mathbb{G}$ (as in [19]), which is not compatible with the revocation mechanism.

Sahai proofs [36] so as to work in the standard model.

In [21], Brickell considered the notion of *verifier-local revocation* group signatures, for which formal definitions were given by Boneh and Shacham [17] and other extensions were proposed in [50, 61, 45]. In this approach, revocation messages are only sent to verifiers and the signing algorithm is completely independent of the number of revocations. Verifiers take as additional input a revocation list (RL), maintained by the group manager, and have to perform a revocation test for each RL entry in order to be convinced that signatures were not issued by a revoked member (a similar revocation mechanism is used in [22]). The verification cost is thus inevitably linear in the number of expelled users.

In 2009, Nakanishi, Fuji, Hira and Funabiki [49] came up with a revocable group signature with constant complexities for signing/verifying. At the same time, group members never have to update their keys. On the other hand, their proposal suffers from linear-size group public keys in the maximal number N of users, although a variant reduces the group public key size to $O(N^{1/2})$.

In anonymous credentials, Tsang *et al.* [57, 58] showed how to prevent users from anonymously authenticating themselves without compromising their anonymity or involving a trusted third party. Their schemes either rely on accumulators (which may be problematic in our setting) or have linear proving complexity in the number of revocations. Camenisch, Kohlweiss and Soriente [25] dealt with revocations in anonymous credentials by periodically updating users credentials in which a specific attribute indicates a validity period. In group signatures, their technique would place an important burden on the group manager who would have to generate updates for each unrevoked individual credential.

While, for various reasons, none of the above constructions conveniently supports large groups, a highly scalable revocation mechanism borrowed from the literature on broadcast encryption was recently described by Libert, Peters and Yung [47] (LPY). Using the Subset Cover framework of Naor, Naor and Lotspiech [51] (NNL), they described a history-independent revocable group signature in the standard model with constant verification time and at most polylogarithmic complexity in other parameters. The technique of [47] blends well with structure-preserving signatures [1, 2] and the Groth-Sahai proofs [36]. The best tradeoff of [47] builds on the Subset Difference (SD) method [51] in its public-key variant due to Dodis and Fazio [31]. It features constant signature size and verification time, $O(\log N)$-size group public keys, revocation lists of size $O(r)$ (as in standard PKIs and group signatures with verifier-local revocation) and membership certificates of size $O(\log^3 N)$. This can be reduced to $O(\log N)$ using the Complete Subtree method [51] but revocation lists are then inflated by a factor of $O(\log N/r)$. Although the Layered Subset Difference method [37] allows for noticeable improvements, the constructions of [47] suffer from relatively large membership certificates. However, some logarithmic dependency on the group size is expected when basing revocation on a tree-like NNL methodology.

1.2 Our Contributions

As mentioned above, to date, in the only scalable revocable group signatures with constant verification time in the standard model [47], group members have to store a polylogarithmic number of group elements. In many applications, however, this can rapidly become unwieldy even for moderately large groups: for example, using the Subset Difference method with $N = 1000 \approx 2^{10}$, users may have to privately store thousands of group elements. In order to be competitive with other group signatures in the standard model such as [35] and still be able to revoke members while keeping them "stateless", it is highly desirable to reduce this complexity.

In this paper, we start with the approach of [47] so as to instantiate the Subset Difference method, but obtain private keys of *constant* size without degrading other performance criteria. This may sound somewhat surprising since, in the SD method, (poly)logarithmic complexities inherently seem inevitable in several metrics. Indeed, in the context of broadcast encryption [51], it requires private keys of size $O(\log^2 N)$ (and even $O(\log^3 N)$ in the public key setting [31] if the result of Boneh-Boyen-Goh [13] is used). Here, we reduce this overhead to a constant while the only dependency on N is a $O(\log N)$-size group public key.

The key idea is as follows. As in the NNL framework, group members are assigned to a leaf of a binary tree and each unrevoked member should belong to exactly one subset in the cover of authorized leafs determined by the group manager. Instead of relying on hierarchical identity-based encryption [15, 38, 33] as in the public-key variant [31] of NNL, we use a novel way for users to non-interactively prove their membership of some generic subset of the SD method using a proof of constant size.

To construct these "compact anonymous membership proofs", we employ *concise* vector commitment schemes [46, 27], where each commitment can be opened w.r.t. individual coordinates in a space-efficient manner (namely, the size of a coordinate-wise opening does not depend on the length of the vector). These vector commitments interact nicely with the specific shape of subsets – as differences between two subtrees – in the SD method. Using them, we compactly encode as a vector the path from the user's leaf to the root. To provide evidence of their inclusion in one of the SD subsets, group members successively prove the equality and the inequality between two coordinates of their vector (*i.e.*, two nodes of the path from their leaf to the root) and specific node labels indicated by an appropriate entry of the revocation list. This is where the position-wise openability of concise commitments is very handy. Of course, for anonymity purposes, the relevant entry of the revocation list only appears in committed form in the group signature. In order to prove that he is using a legal entry of the revocation list, the user generates a set membership proof [23] and proves knowledge of a signature from the group manager on the committed RL entry.

Our technique allows making the most of the LPY approach [47] by reducing the size of membership certificates to a small constant: at the cost of lengthening signatures by a factor of only 1.5, we obtain membership certificates consisting of only 9 group elements and a small integer. For $N = 1000$, users' private keys are thus compressed by a multiplicative factor of several hundreds and this can only become more dramatic for larger groups. At the same time, our main scheme retains all the useful properties of [47]: like the construction of Nakanishi *et al.* [49], it does not require users to update their membership certificates at any time but, unlike [49], our group public key size is $O(\log N)$. Like the SD-based construction of [47], our system uses revocation lists of size $O(r)$, which is on par with Certificate Revocation Lists (CRLs) of standard PKIs. It is worth noting that RLs are *not* part of the group public key: verifiers only need to know the number of the latest revocation epoch and they should not bother to read RLs entirely.

Eventually, our novel approach yields revocable group signatures that become competitive with the regular CRL approach in PKIs: signature generation and verification have constant cost, signatures and membership certificates being of $O(1)$-size while revocation lists have size $O(r)$. A detailed efficiency comparison with previous approaches is given in Section 4. Finally, it is conceivable that our improved revocation technique can find applications beyond group signatures.

2 Background

2.1 Bilinear Maps and Complexity Assumptions

We use bilinear maps $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ over groups of prime order p where $e(g, h) \neq 1_{\mathbb{G}_T}$ if and only if $g, h \neq 1_{\mathbb{G}}$. In these groups, we rely on hardness assumptions that are all falsifiable [52].

Definition 1 ([14]). *The* **Decision Linear Problem** *(DLIN) in $\mathbb{G}$, is to distinguish the distributions $(g^a, g^b, g^{ac}, g^{bd}, g^{c+d})$ and $(g^a, g^b, g^{ac}, g^{bd}, g^z)$, with $a, b, c, d \xleftarrow{R} \mathbb{Z}_p^*$, $z \xleftarrow{R} \mathbb{Z}_p^*$. The* **Decision Linear Assumption** *is the intractability of DLIN for any PPT distinguisher* D.

Definition 2 ([12]). *The q-***Strong Diffie-Hellman problem** *(q-SDH) in $\mathbb{G}$ is, given a tuple $(g, g^a, \ldots, g^{(a^q)})$, for some $g \xleftarrow{R} \mathbb{G}$ and $a \xleftarrow{R} \mathbb{Z}_p$, to find a pair $(g^{1/(a+s)}, s) \in \mathbb{G} \times \mathbb{Z}_p$.*

We use a signature scheme proposed by Abe *et al.* [1], the security of which relies on this assumption.

Definition 3 ([1]). *In a group $\mathbb{G}$, the q-***Simultaneous Flexible Pairing Problem** *(q-SFP) is, given $\big(g_z,\ h_z,\ g_r,\ h_r,\ a,\ \tilde{a},\ b,\ \tilde{b} \in \mathbb{G}\big)$ and $q \in \mathsf{poly}(\lambda)$ tuples $(z_j, r_j, s_j, t_j, u_j, v_j, w_j) \in \mathbb{G}^7$ such that*

$$e(a, \tilde{a}) = e(g_z, z_j) \cdot e(g_r, r_j) \cdot e(s_j, t_j) \quad \text{and} \quad e(b, \tilde{b}) = e(h_z, z_j) \cdot e(h_r, u_j) \cdot e(v_j, w_j), \tag{1}$$

to find a tuple $(z^\star, r^\star, s^\star, t^\star, u^\star, v^\star, w^\star) \in \mathbb{G}^7$ satisfying relation (1) and such that $z^\star \notin \{1_{\mathbb{G}}, z_1, \ldots, z_q\}$.

The paper will appeal to two other assumptions. The first one was implicitly introduced in [16].

Definition 4 ([16]). *Let $\mathbb{G}$ be a group of prime order p. The ℓ-***Diffie-Hellman Exponent** *(ℓ-DHE) problem is, given elements $(g, g_1, \ldots, g_\ell, g_{\ell+2}, \ldots, g_{2\ell}) \in \mathbb{G}^{2\ell}$ such that $g_i = g^{(\alpha^i)}$ for each i and where $\alpha \xleftarrow{R} \mathbb{Z}_p^*$, to compute the missing element $g_{\ell+1} = g^{(\alpha^{\ell+1})}$.*

We actually need a stronger variant, used in [39], of the ℓ-DHE assumption. The Flexible Diffie-Hellman assumption [43] asserts the hardness of finding a non-trivial triple $(g^\mu, g^{a \cdot \mu}, g^{ab \cdot \mu})$, for some non-zero $\mu \in \mathbb{Z}_p^*$, given (g, g^a, g^b). The following assumption relaxes the ℓ-DHE assumption in a similar way.

Definition 5. *In a group $\mathbb{G}$ of prime order p, the* **Flexible ℓ-Diffie-Hellman Exponent** *(ℓ-FlexDHE) problem is, given $(g, g_1, \ldots, g_\ell, g_{\ell+2}, \ldots, g_{2\ell}) \in \mathbb{G}^{2\ell}$ such that $g_i = g^{(\alpha^i)}$ for each i and where $\alpha \xleftarrow{R} \mathbb{Z}_p^*$, to compute a non-trivial triple $(g^\mu, g_{\ell+1}^\mu, g_{2\ell}^\mu) \in (\mathbb{G} \backslash \{1_{\mathbb{G}}\})^3$, for some $\mu \in \mathbb{Z}_p^*$ and where $g_{\ell+1} = g^{(\alpha^{\ell+1})}$.*

The reason why we need to rely on the above assumption instead of the weaker ℓ-DHE assumption is that, in our proofs, the exponent $\mu \in \mathbb{Z}_p$ will appear inside Groth-Sahai commitments [36], from which only values of the form $(g^\mu, g_{\ell+1}^\mu)$ will be efficiently extractable. The additional element $g_{2\ell}^\mu$ will thus prevent the adversary from simply choosing $\mu = \alpha$ or $\mu = \alpha^{-1}$.

A proof of the generic hardness of the ℓ-FlexDHE problem is given in [39]. We note that, while the strength of the assumption grows with ℓ, ℓ is only logarithmic in the maximal number of users here.

2.2 Groth-Sahai Proof Systems

The fundamental Groth-Sahai (GS) techniques [36] can be based on the DLIN assumption, where they use prime order groups and a common reference string containing three vectors $\vec{f_1}, \vec{f_2}, \vec{f_3} \in \mathbb{G}^3$, where $\vec{f_1} = (f_1, 1, g)$, $\vec{f_2} = (1, f_2, g)$ for some $f_1, f_2 \in \mathbb{G}$. To commit to a group element $X \in \mathbb{G}$, one chooses $r, s, t \xleftarrow{R} \mathbb{Z}_p^*$ and computes $\vec{C} = (1, 1, X) \cdot \vec{f_1}^r \cdot \vec{f_2}^s \cdot \vec{f_3}^t$. In the perfect soundness setting, we have $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$ where $\xi_1, \xi_2 \in \mathbb{Z}_p^*$. Commitments $\vec{C} = (f_1^{r+\xi_1 t}, f_2^{s+\xi_2 t}, X \cdot g^{r+s+t(\xi_1+\xi_2)})$ are then extractable (and distributed as Boneh-Boyen-Shacham (BBS) ciphertexts [14]) using $\beta_1 = \log_g(f_1)$, $\beta_2 = \log_g(f_2)$. In the witness indistinguishability (WI) setting, vectors $\vec{f_1}, \vec{f_2}, \vec{f_3}$ 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 an exponent $x \in \mathbb{Z}_p$, one computes $\vec{C} = \vec{\varphi}^x \cdot \vec{f_1}^r \cdot \vec{f_2}^s$, where $r, s \xleftarrow{R} \mathbb{Z}_p^*$, using a CRS consisting of vectors $\vec{\varphi}, \vec{f_1}, \vec{f_2}$. In the perfect soundness setting, $\vec{\varphi}, \vec{f_1}, \vec{f_2}$ are linearly independent ($\vec{\varphi}$ is often chosen as $\vec{\varphi} = \vec{f_3} \cdot (1, 1, g)$, where $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$, for example) whereas, in the WI setting, choosing $\vec{\varphi} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$ gives a perfectly hiding commitment since $\vec{C}$ is always a BBS encryption of $1_{\mathbb{G}}$.

To prove that committed variables satisfy a set of relations, the prover computes one commitment per variable and one proof element per relation. Such non-interactive witness indistinguishable (NIWI) proofs are available for pairing-product equations, which are relations of the type

$$\prod_{i=1}^{n} e(\mathcal{A}_i, \mathcal{X}_i) \cdot \prod_{i=1}^{n} \cdot \prod_{j=1}^{n} e(\mathcal{X}_i, \mathcal{X}_j)^{a_{ij}} = t_T, \tag{2}$$

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

$$\prod_{i=1}^{m} \mathcal{A}_i^{y_i} \cdot \prod_{j=1}^{n} \mathcal{X}_j^{b_j} \cdot \prod_{i=1}^{m} \cdot \prod_{j=1}^{n} \mathcal{X}_j^{y_i \gamma_{ij}} = T, \tag{3}$$

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 pairing-product equations, proofs for quadratic equations consist of 9 group elements whereas linear equations (*i.e.*, where $a_{ij} = 0$ for all i, j in equation (2)) only demand 3 group elements each. Linear multi-exponentiation equations of the type $\prod_{i=1}^{m} \mathcal{A}_i^{y_i} = T$ demand 2 group elements.

Multi-exponentiation equations admit zero-knowledge (NIZK) proofs at no additional cost. On a simulated CRS (prepared for the WI setting), a trapdoor allows simulating proofs without using the witnesses and simulated proofs are distributed as real proofs.

2.3 Structure-Preserving Signatures

Many anonymity-related protocols (e.g., [28, 1, 2, 32, 3]) require to sign elements of bilinear groups while maintaining the feasibility of conveniently proving that a committed signature is valid for a committed message.

Abe, Haralambiev and Ohkubo [1, 2] (AHO) showed how to sign messages of n group elements using signatures consisting of $O(1)$ group elements. In the context of symmetric pairings, the description hereafter assumes public parameters $\mathsf{pp} = \big((\mathbb{G}, \mathbb{G}_T),\ g\big)$ consisting of groups $(\mathbb{G}, \mathbb{G}_T)$ of

order $p > 2^\lambda$, where $\lambda \in \mathbb{N}$ is a security parameter, with a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ and a generator $g \in \mathbb{G}$.

Keygen(pp, n)**:** given an upper bound $n \in \mathbb{N}$ on the number of group elements per signed message, choose generators $G_r, H_r \xleftarrow{R} \mathbb{G}$. Pick $\gamma_z, \delta_z \xleftarrow{R} \mathbb{Z}_p$ and $\gamma_i, \delta_i \xleftarrow{R} \mathbb{Z}_p$, for $i = 1$ to n. Then, compute $G_z = G_r^{\gamma_z}$, $H_z = H_r^{\delta_z}$ and $G_i = G_r^{\gamma_i}$, $H_i = H_r^{\delta_i}$ for each $i \in \{1, \ldots, n\}$. Finally, choose $\alpha_a, \alpha_b \xleftarrow{R} \mathbb{Z}_p$ and define $A = e(G_r, g^{\alpha_a})$ and $B = e(H_r, g^{\alpha_b})$. The public key is defined to be

$$pk = \big(G_r,\ H_r,\ G_z,\ H_z,\ \{G_i, H_i\}_{i=1}^n,\ A,\ B\big) \in \mathbb{G}^{2n+4} \times \mathbb{G}_T^2$$

while the private key is $sk = \big(\alpha_a, \alpha_b, \gamma_z, \delta_z, \{\gamma_i, \delta_i\}_{i=1}^n\big)$.

Sign$(sk, (M_1, \ldots, M_n))$**:** to sign a vector $(M_1, \ldots, M_n) \in \mathbb{G}^n$ using $sk = (\alpha_a, \alpha_b, \gamma_z, \delta_z, \{\gamma_i, \delta_i\}_{i=1}^n)$, choose $\zeta, \rho_a, \rho_b, \omega_a, \omega_b \xleftarrow{R} \mathbb{Z}_p$ and compute $\theta_1 = g^\zeta$ as well as

$$\theta_2 = g^{\rho_a - \gamma_z \zeta} \cdot \prod_{i=1}^n M_i^{-\gamma_i}, \qquad \theta_3 = G_r^{\omega_a}, \qquad \theta_4 = g^{(\alpha_a - \rho_a)/\omega_a},$$
$$\theta_5 = g^{\rho_b - \delta_z \zeta} \cdot \prod_{i=1}^n M_i^{-\delta_i}, \qquad \theta_6 = H_r^{\omega_b}, \qquad \theta_7 = g^{(\alpha_b - \rho_b)/\omega_b},$$

The signature consists of $\sigma = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7)$.

Verify$(pk, \sigma, (M_1, \ldots, M_n))$**:** parse σ as $(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \in \mathbb{G}^7$ and return 1 iff these equalities hold:

$$A = e(G_z, \theta_1) \cdot e(G_r, \theta_2) \cdot e(\theta_3, \theta_4) \cdot \prod_{i=1}^n e(G_i, M_i),$$
$$B = e(H_z, \theta_1) \cdot e(H_r, \theta_5) \cdot e(\theta_6, \theta_7) \cdot \prod_{i=1}^n e(H_i, M_i).$$

The scheme was proved [1, 2] existentially unforgeable under chosen-message attacks under the q-SFP assumption, where q is the number of signing queries.

Signatures can be publicly randomized to obtain a different signature $\{\theta_i'\}_{i=1}^7 \leftarrow \mathsf{ReRand}(pk, \sigma)$ on $(M_1, \ldots, M_n)$. After randomization, we have $\theta_1' = \theta_1$ whereas other signature components $\{\theta_i'\}_{i=2}^7$ are uniformly distributed among the values satisfying $e(G_r, \theta_2') \cdot e(\theta_3', \theta_4') = e(G_r, \theta_2) \cdot e(\theta_3, \theta_4)$ and $e(H_r, \theta_5') \cdot e(\theta_6', \theta_7') = e(H_r, \theta_5) \cdot e(\theta_6, \theta_7)$. Moreover, $\{\theta_i'\}_{i \in \{3,4,6,7\}}$ are statistically independent of the message and the rest of the signature. This implies that, in privacy-preserving protocols, re-randomized $\{\theta_i'\}_{i \in \{3,4,6,7\}}$ can be safely given in the clear as long as $(M_1, \ldots, M_n)$ and $\{\theta_i'\}_{i \in \{1,2,5\}}$ are given in committed form.

In [3], Abe, Groth, Haralambiev and Ohkubo described a more efficient structure-preserving signature based on interactive assumptions. Here, we only rest on non-interactive assumptions.

2.4 Vector Commitment Schemes

We use concise vector commitment schemes, where commitments can be opened with a short decommitment string for each individual coordinate. Such commitments based on ideas from [16, 24]

were described by Libert and Yung [46] and, under weaker assumptions, by Catalano and Fiore [27]. In [46], the commitment key is $ck = (g, g_1, \ldots, g_\ell, g_{\ell+2}, \ldots, g_{2\ell}) \in \mathbb{G}^{2\ell}$, where $g_i = g^{(\alpha^i)}$ for each i. The trapdoor of the commitment is $g_{\ell+1}$, which does not appear in ck. To commit to a vector $\vec{m} = (m_1, \ldots, m_\ell)$, the committer picks $r \xleftarrow{R} \mathbb{Z}_p$ and computes $C = g^r \cdot \prod_{\kappa=1}^{\ell} g_{\ell+1-\kappa}^{m_\kappa}$. A single group element $W_i = g_i^r \cdot \prod_{\kappa=1,\kappa\neq i}^{\ell} g_{\ell+1-\kappa+i}^{m_\kappa}$ provides evidence that m_i is the i-th component of $\vec{m}$ as it satisfies the relation $e(g_i, C) = e(g, W_i) \cdot e(g_1, g_\ell)^{m_i}$. The infeasibility of opening a commitment to two distinct messages for some coordinate i relies on the ℓ-DHE assumption. For our purposes, we only rely on the position-wise binding property of vector commitments and do not need them to be hiding. The randomizer r will thus be removed from the expression of C.

2.5 The NNL Framework for Broadcast Encryption

The important Subset Cover framework [51] considers secret-key broadcast encryption schemes with $N = 2^\ell$ registered receivers. Each receiver is associated with a leaf of a complete binary tree T of height ℓ where each node is assigned a secret key. If $\mathcal{N}$ denotes the universe of users and $\mathcal{R} \subset \mathcal{N}$ is the set of revoked receivers, the framework's idea is to partition the set of non-revoked users into m disjoint subsets $S_1, \ldots, S_m$ such that $\mathcal{N}\backslash\mathcal{R} = S_1 \cup \ldots \cup S_m$. Depending on the way to divide $\mathcal{N}\backslash\mathcal{R}$, different tradeoffs are possible.

The Subset Difference (SD) method yields a transmission cost of $O(|\mathcal{R}|)$ and a storage complexity in $O(\log^2 N)$. For each node $x_j \in \mathsf{T}$, we call T_{x_j} the subtree rooted at x_j. The unrevoked set $\mathcal{N}\backslash\mathcal{R}$ is partitioned into disjoint subsets $S_{k_1,u_1}, \ldots, S_{k_m,u_m}$. For each $i \in \{1, \ldots, m\}$, the subset S_{k_i,u_i} is determined by a node x_{k_i} and one of its descendants x_{u_i} – which are called *primary* and *secondary* roots of S_{k_i,u_i}, respectively – and it consists of the leaves of $\mathsf{T}_{x_{k_i}}$ that are not in $\mathsf{T}_{x_{u_i}}$. Each user belongs to many generic subsets, so that the number of subsets bounded by $m = 2 \cdot |\mathcal{R}| - 1$, as proved in [51].

In the broadcast encryption scenario, a sophisticated key distribution process is necessary to avoid a prohibitive storage overhead. Each subset S_{k_i,u_i} is assigned a "proto-key" $P_{x_{k_i},x_{u_i}}$ that allows deriving the actual symmetric encryption key K_{k_i,u_i} for S_{k_i,u_i} and as well as proto-keys $P_{x_{k_i},x_{u_l}}$ for any descendant x_{u_l} of x_{u_i}. Eventually, each user has to store $O(\log^2 N)$ keys. In the setting of group signatures, we will show that, somewhat unexpectedly, the use of vector commitment schemes allows reducing the private storage to a constant: the size of users' private keys only depends on the security parameter λ, and not on N.

2.6 Revocable Group Signatures

As in [49, 47] (and w.l.o.g.), we consider schemes that have their lifetime divided into revocation epochs at the beginning of which group managers update their revocation lists.

The syntax and the security model are similar to those used by Kiayias and Yung [40]. Like the Bellare-Shi-Zhang model [10], the Kiayias-Yung model assumes an interactive join protocol whereby the user becomes a group member by interacting with the group manager.

Syntax. We denote by $N \in \mathsf{poly}(\lambda)$ the maximal number of group members. At the beginning of each revocation epoch t, the group manager publicizes an up-to-date revocation list RL_t and we denote by $\mathcal{R}_t \subset \{1, \ldots, N\}$ the corresponding set of revoked users (we assume that $\mathcal{R}_t$ is part of RL_t). A revocable group signature (R-GS) scheme consists of the following algorithms or protocols.

Setup(λ, N)**:** given a security parameter $\lambda \in \mathbb{N}$ and a maximal number of group members $N \in \mathbb{N}$, this algorithm (which is run by some trusted party) generates a group public key $\mathcal{Y}$, the group manager's private key $\mathcal{S}_{\mathsf{GM}}$ and the opening authority's private key $\mathcal{S}_{\mathsf{OA}}$. Keys $\mathcal{S}_{\mathsf{GM}}$ and $\mathcal{S}_{\mathsf{OA}}$ are given to the appropriate authority while $\mathcal{Y}$ is publicized. The algorithm also initializes a public state St comprising a set data structure $St_{users} = \emptyset$ and a string data structure $St_{\mathsf{trans}} = \epsilon$.

Join: is an interactive protocol between the group manager GM and a prospective group member $\mathcal{U}_i$. The protocol involves two interactive Turing machines $\mathsf{J}_{\mathsf{user}}$ and J_{GM} that both take as input $\mathcal{Y}$. The execution, denoted as $[\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y}), \mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})]$, ends with $\mathcal{U}_i$ obtaining a membership secret sec_i, that no one else knows, and a membership certificate cert_i. If the protocol is successful, the group manager updates the public state St by setting $St_{users} := St_{users} \cup \{i\}$ as well as $St_{\mathsf{trans}} := St_{\mathsf{trans}} || \langle i, \mathsf{transcript}_i \rangle$.

Revoke: is a (possibly probabilistic) algorithm allowing the GM to generate an updated revocation list RL_t for the new revocation epoch t. It takes as input a public key $\mathcal{Y}$ and a set $\mathcal{R}_t \subset St_{users}$ that identifies the users to be revoked. It outputs an updated revocation list RL_t for epoch t.

Sign: given a revocation epoch t with its revocation list RL_t, a membership certificate cert_i, a membership secret sec_i and a message M, this algorithm outputs $\perp$ if $i \in \mathcal{R}_t$ and a signature σ otherwise.

Verify: given a signature σ, a revocation epoch t, the corresponding revocation list RL_t, a message M and a group public key $\mathcal{Y}$, this deterministic algorithm returns either 0 or 1.

Open: takes as input a message M, a valid signature σ w.r.t. $\mathcal{Y}$ for the indicated revocation epoch t, the opening authority's private key $\mathcal{S}_{\mathsf{OA}}$ and the public state St. It outputs $i \in St_{users} \cup \{\perp\}$, which is the identity of a group member or a symbol indicating an opening failure.

Each membership certificate contains a unique tag that identifies the user.

A R-GS scheme must satisfy three security notions defined in Appendix A. The first one is called *security against misidentification attacks*. It requires that, even if the adversary can introduce and revoke users at will, it cannot produce a signature that traces outside the set of unrevoked adversarially-controlled users.

As in ordinary (*i.e.*, non-revocable) group signatures, the notion of *security against framing attacks* captures that under no circumstances should an honest user be held accountable for messages that he did not sign, even if the whole system conspires against that user. Finally, the notion of *anonymity* is also defined (by granting the adversary access to a signature opening oracle) as in the models of [10, 40].

3 A Revocable Group Signature with Compact Keys and Constant Verification Time

The number of users is assumed to be $N = 2^{\ell-1} \in \mathsf{poly}(\lambda)$, for some integer ℓ, so that each group member is assigned to a leaf of the tree. Each node is assigned a unique identifier. For simplicity, the root is identified by $\mathsf{ID}(\epsilon) = 1$ and, for each other node x, we define the identifier $\mathsf{ID}(x) \in \{1, \ldots, 2N-1\}$ to be $\mathsf{ID}(x) = 2 \cdot \mathsf{ID}(\mathsf{parent}(x)) + b$, where $\mathsf{parent}(x)$ denotes x's father in the tree and $b = 0$ (resp. $b = 1$) if x is the left (resp. right) child of its father. The root of the tree is assigned the identifier $\mathsf{ID}_\epsilon = 1$.

At the beginning of each revocation epoch t, the GM generates an up-to-date revocation list RL_t containing one entry for each generic subset $S_{k_1,u_1}, \ldots, S_{k_m,u_m}$ produced by the Subset Difference method. These subsets are encoded in such a way that unrevoked users can anonymously prove

their membership of one of them. Our technique allows to do this using a proof of *constant* size.

The intuition is as follows. In the generation of RL_t, for each $i \in \{1, \ldots, m\}$, if x_{k_i} (resp. x_{u_i}) denotes the primary (resp. secondary) root of S_{k_i,u_i}, the GM encodes S_{k_i,u_i} as a vector of group elements R_i that determines the levels of nodes x_{k_i} and x_{u_i} in the tree (which are called ϕ_i and ψ_i hereafter) and the identifiers $\mathsf{ID}(x_{k_i})$ and $\mathsf{ID}(x_{u_i})$. Then, the resulting vector R_i is authenticated by means of a structure preserving signature Θ_i, which is included in RL_t and will be used in a set membership proof [23].

During the join protocol, users obtain from the GM a structure-preserving signature on a compact encoding C_v – which is computed as a commitment to a vector of node identifiers $(I_1, \ldots, I_\ell)$ – of the path $(I_1, \ldots, I_\ell)$ between their leaf v and the root ϵ. This path is encoded as a single group element.

In order to anonymously prove his non-revocation, a group member $\mathcal{U}_i$ uses RL_t to determine the generic subset S_{k_l,u_l}, with $l \in \{1, \ldots, m\}$, where his leaf v_i lies. He commits to the corresponding vector of group elements R_l that encodes the node identifiers $\mathsf{ID}(x_{k_l})$ and $\mathsf{ID}(x_{u_l})$ of the primary and secondary roots of S_{k_l,u_l} at levels ϕ_l and ψ_l, respectively. If $(I_1, \ldots, I_\ell)$ identifies the path from his leaf v_i to ϵ, the unrevoked member $\mathcal{U}_i$ generates a membership proof for the subset S_{k_l,u_l} by proving that $\mathsf{ID}(x_{k_l}) = I_{\phi_l}$ and $\mathsf{ID}(x_{u_l}) \neq I_{\psi_l}$ (in other words, that x_{k_l} is an ancestor of v_i and x_{u_l} is not). To succinctly prove these statements, $\mathcal{U}_i$ uses the properties of the vector commitment scheme recalled in Section 2.4. Finally, in order to convince the verifier that he used a legal element of RL_t, $\mathcal{U}_i$ follows the technique of [23] and proves knowledge of a signature Θ_l on the committed vector of group elements R_l. By doing so, $\mathcal{U}_i$ thus provides evidence that his leaf v_i is a member of some authorized subset S_{k_l,u_l} without revealing $l \in \{1, \ldots, m\}$.

In order to obtain the strongest flavor of anonymity (*i.e.*, where the adversary has access to a signature opening oracle), the scheme uses Kiltz's tag-based encryption scheme [42] as in Groth's construction [35]. In non-frameability concerns, the group member $\mathcal{U}_i$ also generates a weak Boneh-Boyen signature [12] (which yields a fully secure signature when combined with a one-time signature) using $x = \log_g(X)$, where $X \in \mathbb{G}$ is a group element certified by the GM and bound to the path $(I_1, \ldots, I_\ell)$ during the join protocol.

3.1 Construction

As in standard security models for group signatures, we assume that, before joining the group, user $\mathcal{U}_i$ chooses a long term key pair $(\mathsf{usk}[i], \mathsf{upk}[i])$ and registers it in some PKI.

Setup(λ, N)**:** given a security parameter $\lambda \in \mathbb{N}$ and the permitted number of users $N = 2^{\ell-1}$,

1. Choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$, with a generator $g \xleftarrow{R} \mathbb{G}$.
2. Define $n_0 = 2$ and $n_1 = 5$. Generate two key pairs $(sk^{(0)}_{\mathsf{AHO}}, pk^{(0)}_{\mathsf{AHO}})$ and $(sk^{(1)}_{\mathsf{AHO}}, pk^{(1)}_{\mathsf{AHO}})$ for the AHO signature in order to sign messages of n_0 and n_1 group elements, respectively. These key pairs are

$$pk^{(d)}_{\mathsf{AHO}} = \Big(G_r^{(d)},\ H_r^{(d)},\ G_z^{(d)} = G_r^{\gamma_z^{(d)}},\ H_z^{(d)} = H_r^{\delta_z^{(d)}},\ \{G_i^{(d)} = G_r^{\gamma_i^{(d)}}, H_i^{(d)} = H_r^{\delta_i^{(d)}}\}_{i=1}^{n_d},\ A^{(d)},\ B^{(d)}\Big)$$

   and $sk^{(d)}_{\mathsf{AHO}} = \big(\alpha_a^{(d)}, \alpha_b^{(d)}, \gamma_z^{(d)}, \delta_z^{(d)}, \{\gamma_i^{(d)}, \delta_i^{(d)}\}_{i=1}^{n_d}\big)$, where $d \in \{0, 1\}$. These two schemes will be used to sign messages consisting of 2 and 5 group elements, respectively.

3. Generate a public key $ck = (g_1, \ldots, g_\ell, g_{\ell+2}, \ldots, g_{2\ell}) \in \mathbb{G}^{2\ell-1}$ for vectors of dimension ℓ in the vector commitment scheme recalled in section 2.4. The trapdoor $g_{\ell+1}$ is not needed and can be discarded.
4. As a CRS for the NIWI proof system, select vectors $\mathbf{f} = (\vec{f_1}, \vec{f_2}, \vec{f_3})$ s.t. $\vec{f_1} = (f_1, 1, g) \in \mathbb{G}^3$, $\vec{f_2} = (1, f_2, g) \in \mathbb{G}^3$, and $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$, with $f_1 = g^{\beta_1}, f_2 = g^{\beta_2} \xleftarrow{R} \mathbb{G}$ and $\beta_1, \beta_2, \xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$. We also define the vector $\vec{\varphi} = \vec{f_3} \cdot (1, 1, g)$.
5. Choose $(U, V) \xleftarrow{R} \mathbb{G}^2$ that, together with generators $f_1, f_2, g \in \mathbb{G}$, will form a public encryption key.
6. Select a strongly unforgeable one-time signature $\Sigma = (\mathcal{G}, \mathcal{S}, \mathcal{V})$.
7. Set $\mathcal{S}_{\mathsf{GM}} := \big(sk^{(0)}_{\mathsf{AHO}}, sk^{(1)}_{\mathsf{AHO}}\big)$, $\mathcal{S}_{\mathsf{OA}} := \big(\beta_1, \beta_2\big)$ as authorities' private keys and the group public key is

$$\mathcal{Y} := \Big(g,\ pk^{(0)}_{\mathsf{AHO}},\ pk^{(1)}_{\mathsf{AHO}},\ ck = (g_1, \ldots, g_\ell, g_{\ell+2}, \ldots, g_{2\ell}),\ \mathbf{f},\ \vec{\varphi},\ (U, V),\ \Sigma\Big).$$

Join$^{(\mathrm{GM},\mathcal{U}_i)}$**:** the group manager and the prospective user $\mathcal{U}_i$ run the following interactive protocol $[\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y}), \mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})]$:

1. $\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y})$ draws $x \xleftarrow{R} \mathbb{Z}_p$ and computes $X = g^x$ which is sent to $\mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})$. If $X \in \mathbb{G}$ already appears in some entry $\mathsf{transcript}_j$ of the database St_{trans}, J_{GM} halts and returns $\perp$ to $\mathsf{J}_{\mathsf{user}}$.
2. J_{GM} assigns to $\mathcal{U}_i$ an available leaf v of identifier $\mathsf{ID}(v)$ in the tree T. Let $x_1, \ldots, x_\ell$ be the path from $x_\ell = v$ to the root $x_1 = \epsilon$ of T. Let also $(I_1, \ldots, I_\ell) = (\mathsf{ID}(x_1), \ldots, \mathsf{ID}(x_\ell))$ be the corresponding vector of identifiers (with $I_1 = 1$ and $I_\ell = \mathsf{ID}(v) \in \{N, \ldots, 2N-1\}$). Then, J_{GM} does the following.
   a. Compute a compact encoding of $(I_1, \ldots, I_\ell)$ as $C_v = \prod_{\kappa=1}^{\ell} g_{\ell+1-\kappa}^{I_\kappa} = g_\ell^{I_1} \cdots g_1^{I_\ell}$.
   b. Using $sk^{(0)}_{\mathsf{AHO}}$, generate an AHO signature $\sigma_v = (\theta_{v,1}, \ldots, \theta_{v,7})$ on the pair $(X, C_v) \in \mathbb{G}^2$ so as to bind the encoded path C_v to the value X that identifies $\mathcal{U}_i$.
3. J_{GM} sends $\mathsf{ID}(v) \in \{N, \ldots, 2N-1\}$ and C_v to $\mathsf{J}_{\mathsf{user}}$ that halts if $\mathsf{ID}(v) \notin \{N, \ldots, 2N-1\}$ or if C_v is found incorrect. Otherwise, $\mathsf{J}_{\mathsf{user}}$ sends a signature $sig_i = \mathrm{Sign}_{\mathsf{usk}[i]}\big(X||(I_1, \ldots, I_\ell)\big)$ to J_{GM}.
4. J_{GM} checks that $\mathrm{Verify}_{\mathsf{upk}[i]}\big((X||(I_1, \ldots, I_\ell)), sig_i\big) = 1$. If not J_{GM} aborts. Otherwise, J_{GM} returns the AHO signature σ_v to $\mathsf{J}_{\mathsf{user}}$ and stores $\mathsf{transcript}_i = (X, \mathsf{ID}(v), C_v, \sigma_v, sig_i)$ in the database St_{trans}.
5. $\mathsf{J}_{\mathsf{user}}$ defines the membership certificate as $\mathsf{cert}_i = \big(\mathsf{ID}(v), X, C_v, \sigma_v\big) \in \{N, \ldots, 2N-1\} \times \mathbb{G}^9$, where X will serve as the tag identifying $\mathcal{U}_i$. The membership secret sec_i is defined as $\mathsf{sec}_i = x \in \mathbb{Z}_p$.

Revoke$(\mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, t, \mathcal{R}_t)$**:** Parse $\mathcal{S}_{\mathsf{GM}}$ as $\mathcal{S}_{\mathsf{GM}} := \big(sk^{(0)}_{\mathsf{AHO}}, sk^{(1)}_{\mathsf{AHO}}\big)$ and do the following.

1. Using the subset covering algorithm of the SD method, find a cover of the unrevoked user set $\{1, \ldots, N\} \backslash \mathcal{R}_t$ as the union of disjoint subsets of the form $S_{k_1,u_1}, \ldots, S_{k_m,u_m}$, where $m \leq 2 \cdot |\mathcal{R}_t| - 1$.
2. For $i = 1$ to m, do the following.
   a. Consider the subset S_{k_i,u_i} as the difference between sub-trees rooted at an internal node x_{k_i} and one of its descendants x_{u_i}. Let $\phi_i, \psi_i \in \{1, \ldots, \ell\}$ be the depths of x_{k_i} and

x_{u_i}, respectively, in T assuming that the root ϵ is at depth 1. Encode S_{k_i,u_i} as a vector $\big(g_{\phi_i}, g_1^{\mathsf{ID}(x_{k_i})}, g_{\psi_i}, g^{\mathsf{ID}(x_{u_i})}\big)$.

b. To authenticate S_{k_i,u_i} and bind it to the revocation epoch t, use $sk^{(1)}_{\mathsf{AHO}}$ to generate an AHO signature $\Theta_i = (\Theta_{i,1}, \ldots, \Theta_{i,7}) \in \mathbb{G}^7$ on $R_i = \Big(g^t, g_{\phi_i}, g_1^{\mathsf{ID}(x_{k_i})}, g_{\psi_i}, g^{\mathsf{ID}(x_{u_i})}\Big)$, where the epoch number t is interpreted as an element of $\mathbb{Z}_p$.

Return the revocation data

$$RL_t = \Big(t,\ \mathcal{R}_t,\ \{\phi_i,\ \psi_i,\ \mathsf{ID}(x_{k_i}),\ \mathsf{ID}(x_{u_i}),\ \Theta_i = (\Theta_{i,1}, \ldots, \Theta_{i,7})\}_{i=1}^{m}\Big). \tag{4}$$

Sign$(\mathcal{Y}, t, RL_t, \mathsf{cert}_i, \mathsf{sec}_i, M)$**:** return $\perp$ if $i \in \mathcal{R}_t$. Otherwise, to sign $M \in \{0,1\}^*$, generate a one-time signature key pair $(\mathsf{SK}, \mathsf{VK}) \leftarrow \mathcal{G}(\lambda)$. Parse cert_i as $\mathsf{cert}_i = \big(\mathsf{ID}(v_i), X, C_{v_i}, \sigma_{v_i}\big) \in \{N, \ldots, 2N-1\} \times \mathbb{G}^9$ and sec_i as $x \in \mathbb{Z}_p$. Let $\epsilon = x_1, \ldots, x_\ell = v_i$ be the path connecting v_i to the root ϵ of T and let $(I_1, \ldots, I_\ell) = (\mathsf{ID}(x_1), \ldots, \mathsf{ID}(x_\ell))$ be the vector of node identifiers. First, $\mathcal{U}_i$ generates a commitment $com_{C_{v_i}}$ to the encoding C_{v_i} of the path $(I_1, \ldots, I_\ell)$ from v_i to the root. Then, he does the following.

1. Using RL_t, find the set S_{k_l,u_l}, with $l \in \{1, \ldots, m\}$, that contains the leaf v_i identified by $\mathsf{ID}(v_i)$. Let x_{k_l} and x_{u_l} denote the primary and secondary roots of S_{k_l,u_l} at depths ϕ_l and ψ_l, respectively. Since x_{k_l} is an ancestor of v_i but x_{u_l} is not, it must be the case that $I_{\phi_l} = \mathsf{ID}(x_{k_l})$ and $I_{\psi_l} \neq \mathsf{ID}(x_{u_l})$.

2. To prove that v_i belongs to S_{k_l,u_l} without leaking l, $\mathcal{U}_i$ first re-randomizes the l-th AHO signature Θ_l of RL_t as $\{\Theta'_{l,i}\}_{i=1}^{7} \leftarrow \mathsf{ReRand}(pk^{(1)}_{\mathsf{AHO}}, \Theta_l)$. Then, he commits to the l-th revocation message

$$R_l = (R_{l,1}, R_{l,2}, R_{l,3}, R_{l,4}, R_{l,5}) = \big(g^t, g_{\phi_l}, g_1^{\mathsf{ID}(x_{k_l})}, g_{\psi_l}, g^{\mathsf{ID}(x_{u_l})}\big) \tag{5}$$

and its signature $\Theta'_l = (\Theta'_{l,1}, \ldots, \Theta'_{l,7})$ by computing Groth-Sahai commitments $\{com_{R_{l,\tau}}\}_{\tau=2}^{5}$, $\{com_{\Theta'_{l,j}}\}_{j\in\{1,2,5\}}$ to $\{R_{l,\tau}\}_{\tau=2}^{5}$ and $\{\Theta'_{l,j}\}_{j\in\{1,2,5\}}$.

a. To prove that $I_{\phi_l} = \mathsf{ID}(x_{k_l})$, $\mathcal{U}_i$ first computes $W_{\phi_l} = \prod_{\kappa=1,\ \kappa\neq\phi_l}^{\ell} g_{\ell+1-\kappa+\phi_l}^{I_\kappa}$ that satisfies the equality $e(g_{\phi_l}, C_{v_i}) = e(g_1, g_\ell)^{I_{\phi_l}} \cdot e(g, W_{\phi_l})$. Then, $\mathcal{U}_i$ generates a Groth-Sahai commitment $com_{W_{\phi_l}}$ to W_{ϕ_l}. He computes a NIWI proof that committed variables $(R_{l,2}, R_{l,3}, C_{v_i}, W_{\phi_l})$ satisfy

$$e(R_{l,2}, C_{v_i}) = e(R_{l,3}, g_\ell) \cdot e(g, W_{\phi_l}). \tag{6}$$

We denote by π_{eq} the proof for the quadratic equation (6), which requires 9 group elements.

b. To prove that $I_{\psi_l} \neq \mathsf{ID}(x_{u_l})$, $\mathcal{U}_i$ computes $W_{\psi_l} = \prod_{\kappa=1,\ \kappa\neq\psi_l}^{\ell} g_{\ell+1-\kappa+\psi_l}^{I_\kappa}$ that satisfies the equality $e(g_{\psi_l}, C_{v_i}) = e(g_1, g_\ell)^{I_{\psi_l}} \cdot e(g, W_{\psi_l})$. Then, he computes a commitment $com_{W_{\psi_l}}$ to W_{ψ_l} as well as commitments com_{Γ_l} and $\{com_{\Psi_{l,\tau}}\}_{\tau\in\{0,1,2\ell\}}$ to the group elements

$$(\Gamma_l, \Psi_{l,0}, \Psi_{l,1}, \Psi_{l,2\ell}) = \big(g^{1/(I_{\psi_l} - \mathsf{ID}(x_{u_l}))}, g^{I_{\psi_l}}, g_1^{I_{\psi_l}}, g_{2\ell}^{I_{\psi_l}}\big).$$

Then, $\mathcal{U}_i$ provides evidence that committed variables $(R_{l,4}, R_{l,5}, C_{v_i}, \Gamma_l, \Psi_{l,0}, \Psi_{l,1}, \Psi_{l,2\ell})$ satisfy

$$e(R_{l,4}, C_{v_i}) = e(\Psi_{l,1}, g_\ell) \cdot e(g, W_{\psi_l}), \qquad e(\Psi_{l,0}/R_{l,5}, \Gamma_l) = e(g,g) \tag{7}$$
$$e(\Psi_{l,1}, g) = e(g_1, \Psi_{l,0}), \qquad e(\Psi_{l,2\ell}, g) = e(g_{2\ell}, \Psi_{l,0}). \tag{8}$$

We denote this NIWI proof by $\pi_{neq} = (\pi_{neq,1}, \pi_{neq,2}, \pi_{neq,3}, \pi_{neq,4})$. Since the first two equations (7) are quadratic, $\pi_{neq,1}$ and $\pi_{neq,2}$ consist of 9 elements each. The last two equations (8) are linear and both cost 3 elements to prove.

3. $\mathcal{U}_i$ provides evidence that the tuple R_l of (5) is a certified revocation message for epoch t: namely, he computes a NIWI proof π_{R_l} that committed message elements $\{R_{l,\tau}\}_{\tau=2}^{5}$ and signature components $\{\Theta'_{l,j}\}_{j\in\{1,2,5\}}$ satisfy the equations

$$A^{(1)} \cdot e(\Theta'_{l,3}, \Theta'_{l,4})^{-1} \cdot e(G_1^{(1)}, g^t)^{-1} = e(G_z^{(1)}, \Theta'_{l,1}) \cdot e(G_r^{(1)}, \Theta'_{l,2}) \cdot \prod_{\tau=2}^{5} e(G_\tau^{(1)}, R_{l,\tau}), \tag{9}$$
$$B^{(1)} \cdot e(\Theta'_{l,6}, \Theta'_{l,7})^{-1} \cdot e(H_1^{(1)}, g^t)^{-1} = e(H_z^{(1)}, \Theta'_{l,1}) \cdot e(H_r^{(1)}, \Theta'_{l,5}) \cdot \prod_{\tau=2}^{5} e(H_\tau^{(1)}, R_{l,\tau}),$$

Since $\{\Theta'_{l,j}\}_{j\in\{3,4,6,7\}}$ are constants, equations (9) are both linear and thus require 3 elements each. Hence, π_{R_l} takes 6 elements altogether.

4. Let $\sigma_{v_i} = (\theta_{v_i,1}, \ldots, \theta_{v_i,7})$ be the AHO signature on the message (X, C_{v_i}). Set $\{\theta'_{v_i,j}\}_{j=1}^{7} \leftarrow \mathsf{ReRand}(pk_{\mathsf{AHO}}^{(0)}, \sigma_{v_i})$ and generate commitments $\{com_{\theta'_{v_i,j}}\}_{j\in\{1,2,5\}}$ to $\{\theta'_{v_i,j}\}_{j\in\{1,2,5\}}$ as well as a commitment com_X to X. Then, generate a NIWI proof $\pi_{\sigma_{v_i}}$ that committed variables satisfy the verification equations

$$A^{(0)} \cdot e(\theta'_{l,3}, \theta'_{l,4})^{-1} = e(G_z^{(0)}, \theta'_{l,1}) \cdot e(G_r^{(0)}, \theta'_{l,2}) \cdot e(G_1^{(0)}, X) \cdot e(G_2^{(0)}, C_{v_i}),$$
$$B^{(0)} \cdot e(\theta'_{l,6}, \theta'_{l,7})^{-1} = e(H_z^{(0)}, \theta_{l,1}) \cdot e(H_r^{(0)}, \theta'_{l,5}) \cdot e(H_1^{(0)}, X) \cdot e(H_2^{(0)}, C_{v_i})$$

Since these equations are linear, $\pi_{\sigma_{v_i}}$ requires 6 group elements.

5. Using VK as a tag (by first hashing it onto $\mathbb{Z}_p$ in such a way that it can be interpreted as a $\mathbb{Z}_p$ element), compute a tag-based encryption [42] of X by drawing $z_1, z_2 \xleftarrow{R} \mathbb{Z}_p$ and setting $(\Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5) = \big(f_1^{z_1}, f_2^{z_2}, X \cdot g^{z_1+z_2}, (g^{\mathsf{VK}} \cdot U)^{z_1}, (g^{\mathsf{VK}} \cdot V)^{z_2}\big)$.
6. Generate a NIZK proof that $com_X = (1,1,X) \cdot \vec{f}_1^{\,w_{X,1}} \cdot \vec{f}_2^{\,w_{X,2}} \cdot \vec{f}_3^{\,w_{X,3}}$ and $(\Upsilon_1, \Upsilon_2, \Upsilon_3)$ are BBS encryptions of the same value X. If we write $\vec{f}_3 = (f_{3,1}, f_{3,2}, f_{3,3})$, the Groth-Sahai commitment com_X can be written as $(f_1^{w_{X,1}} \cdot f_{3,1}^{w_{X,3}}, f_2^{w_{X,2}} \cdot f_{3,2}^{w_{X,3}}, X \cdot g^{w_{X,1}+w_{X,2}} \cdot f_{3,3}^{w_{X,3}})$, so that we have

$$com_X \cdot (\Upsilon_1, \Upsilon_2, \Upsilon_3)^{-1} = \big(f_1^{\chi_1} \cdot f_{3,1}^{\chi_3},\ f_2^{\chi_2} \cdot f_{3,2}^{\chi_3},\ g^{\chi_1+\chi_2} \cdot f_{3,3}^{\chi_3}\big) \tag{10}$$

with $\chi_1 = w_{X,1} - z_1$, $\chi_2 = w_{X,2} - z_2$, $\chi_3 = w_{X,3}$. Compute $com_{\chi_j} = \vec{\varphi}^{\,\chi_j} \cdot \vec{f}_1^{\,w_{\chi_j,1}} \cdot \vec{f}_2^{\,w_{\chi_j,2}}$, with $w_{\chi_j,1}, w_{\chi_j,2} \xleftarrow{R} \mathbb{Z}_p$ for $j \in \{1,2,3\}$, as commitments to $\{\chi_j\}_{j=1}^{3}$ and generates proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^{3}$ that χ_1, χ_2, χ_3 satisfy the three linear relations (10). The proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^{3}$ cost 2 elements each.

7. Compute a weak Boneh-Boyen signature $\sigma_{\mathsf{VK}} = g^{1/(x+\mathsf{VK})}$ on VK and a commitment $com_{\sigma_{\mathsf{VK}}}$ to σ_{VK}. Then, generate a NIWI proof $\pi_{\sigma_{\mathsf{VK}}} = (\vec{\pi}_{\sigma_{\mathsf{VK}},1}, \vec{\pi}_{\sigma_{\mathsf{VK}},2}, \vec{\pi}_{\sigma_{\mathsf{VK}},3}) \in \mathbb{G}^9$ that committed variables $(\sigma_{\mathsf{VK}}, X) \in \mathbb{G}^2$ satisfy the quadratic equation $e(\sigma_{\mathsf{VK}}, X \cdot g^{\mathsf{VK}}) = e(g,g)$.

8. Compute $\sigma_{ots} = \mathcal{S}(\mathsf{SK}, (M, RL_t, \Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5, \Omega, \mathbf{com}, \mathbf{\Pi}))$ where $\Omega = \{\Theta'_{l,i}, \theta'_{l,i}\}_{i\in\{3,4,6,7\}}$ and

$$\mathbf{com} = \big(com_{C_{v_i}}, com_X, \{com_{R_{l,\tau}}\}_{\tau=2}^{5}, com_{W_{\phi_l}}, com_{W_{\psi_l}}, com_{\Gamma_l}, \{com_{\Psi_{l,\tau}}\}_{\tau\in\{0,1,2\ell\}}, \{com_{\Theta'_{l,j}}\}_{j\in\{1,2,5\}}, \{com_{\theta'_{l,j}}\}_{j\in\{1,2,5\}}, \{com_{\chi_j}\}_{j=1}^{3}, com_{\sigma_{\mathsf{VK}}}\big)$$

$$\mathbf{\Pi} = \big(\pi_{eq}, \pi_{neq}, \pi_{R_l}, \pi_{\sigma_{v_i}}, \{\pi_{eq\text{-}com,j}\}_{j=1}^{3}, , \pi_{\sigma_{\mathsf{VK}}}\big)$$

Return the signature $\sigma = \big(\mathsf{VK}, \Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5, \Omega, \mathbf{com}, \mathbf{\Pi}, \sigma_{ots}\big)$.

Verify$(\sigma, M, t, RL_t, \mathcal{Y})$**:** parse σ as above. If $\mathcal{V}(\mathsf{VK}, (M, RL_t, \Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5, \Omega, \mathbf{com}, \mathbf{\Pi}), \sigma_{ots}) = 0$ or if $(\Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5)$ is not a well-formed tag-based encryption (that is, if $e(\Upsilon_1, g^{\mathsf{VK}} \cdot U) \neq e(f_1, \Upsilon_4)$ or $e(\Upsilon_2, g^{\mathsf{VK}} \cdot V) \neq e(f_2, \Upsilon_5)$), return 0. Then, return 1 if all proofs properly verify. Otherwise, return 0.

Open$(M, t, RL_t, \sigma, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St)$**:** parse σ as above and return $\perp$ if $\mathsf{Verify}(\sigma, M, t, RL_t, \mathcal{Y}) = 0$. Otherwise, given $\mathcal{S}_{\mathsf{OA}} = (\beta_1, \beta_2)$, compute $\tilde{X} = \Upsilon_3 \cdot \Upsilon_1^{-1/\beta_1} \cdot \Upsilon_2^{-1/\beta_2}$. In the database St_{trans}, find a record $\langle i, \mathsf{transcript}_i = (X_i, \mathsf{ID}(v_i), C_{v_i}, \sigma_{v_i}, sig_i)\rangle$ such that $X_i = \tilde{X}$. If no such record exists in St_{trans}, return $\perp$. Otherwise, return i.

At first glance, the variable $\Psi_{l,2\ell}$ and the proof of the second equality (8) may seem unnecessary in step 2.b of the signing algorithm. However, this element plays a crucial role when it comes to prove the security under the ℓ-FlexDHE assumption. Indeed, the proof of security against misidentification attacks (more precisely, the proof of Lemma 1 in Appendix B.1) ceases to go through if we remove $\Psi_{l,2\ell}$ and its corresponding proof.

As far as efficiency goes, each entry of RL_t contains 7 group elements and two node identifiers of $O(\log N)$ bits each. If $\lambda_{\mathbb{G}}$ is the bitlength of a group element, we have $\log N \ll \lambda_{\mathbb{G}}/2$ (since $\lambda \leq \lambda_{\mathbb{G}}$ and N is polynomial), so that the number of bits of RL_t is bounded by $2 \cdot |\mathcal{R}_t| \cdot (7 \cdot \lambda_{\mathbb{G}} + 2\log N + 2\log\log N) < 2 \cdot |\mathcal{R}_t| \cdot (9\lambda_{\mathbb{G}})$ bits. The size of RL_t is thus bounded by that of $18 \cdot |\mathcal{R}_t|$ group elements.

Unlike [47], group members only need to store 9 group elements in their membership certificate. As far as the size of signature goes, **com** and $\mathbf{\Pi}$ require 66 and 60 group elements, respectively. If the one-time signature of [34] is used, VK and σ_{ots} consist of 3 elements of $\mathbb{G}$ and 2 elements of $\mathbb{Z}_p$, respectively. The global size σ amounts to that of 144 group elements, which is about 50% longer than [47]. In comparison with [35] (which does not natively support revocation), signatures are only longer by a factor of 3. At the 128-bit security level, each group element should have a 512-bit representation and a signature takes 9 kB.

Verifying signatures takes constant time. The signer has to compute at most $2\ell = O(\log N)$ exponentiations to obtain W_{ϕ_l} and W_{ψ_l} at the beginning of each revocation epoch. Note that these exponentiations involve short exponents of $O(\log N)$ bits each. Hence, computing W_{ϕ_l} and W_{ψ_l} requires $O(\log^2 N)$ multiplications in $\mathbb{G}$. For this reason, since we always have $\log^2 N \ll \lambda$ (as long as $N \ll 2^{\lambda^{1/2}}$), this cost is dominated by that of a single exponentiation in $\mathbb{G}$.

3.2 Security

From a security point of view, we prove the following theorem in Appendix B.

Theorem 1. *The scheme provides anonymity as well as security against misidentification and framing attacks if the SFP, FlexDHE, SDH and DLIN assumptions all hold in* $\mathbb{G}$.

In comparison with [47], the security proof requires the additional non-standard ℓ-FlexDHE assumption, where $\ell = \log(N)$. In Appendix C, we show how to rest on weaker (and fewer) intractability assumptions if we accept to use a group public key of size $O(\log^2 N)$ while keeping all other complexities unchanged. This construction offers an interesting tradeoff since, in some applications, group public keys of log-squared size are handier to work with than private keys of size $O(\log^3 N)$ as in [47].

Appendix C also explains how to also eliminate the SDH assumption using the technique of Malkin et al. [48]. In this case, an additive factor of $O(\lambda)$ appears in the group public key size because a longer Groth-Sahai CRS must be used. On the other hand, the q-SFP assumption becomes the only assumption of variable size.

4 Efficiency Comparisons

This section compares pairing-based revocable group signatures where group members are stateless and do not update their membership certificate whenever a revocation occurs. Comparisons are given in terms of computational costs and the size (measured by the number of group elements) of public keys, signatures, membership certificates and revocation lists as functions of N, r and, in some cases, the number T of revocation epochs. By "constant", we thus mean that the complexity only depends on the security parameter λ.

Table 1. Comparison between pairing-based revocable group signatures

Schemes	Group public key size	Signature size	Membership certificate size	Revocation list size	Signature cost	Verification cost	Revocation cost	Standard model?
NFHF1 [49]	$O(N)$	$O(1)$	$O(1)$	$O(r)$	$O(1)$	$O(1)$	$O(r)$	✗
NFHF2 [49]	$O(N^{1/2})$	$O(1)$	$O(1)$	$O(r)$	$O(1)$	$O(1)$	$O(r)$	✗
BS [17]	$O(1)$	$O(1)$	$O(1)$	$O(r)$	$O(1)$	$O(r)$	$O(1)$	✗
NF [50]	$O(T)^{\diamondsuit}$	$O(1)$	$O(1)$	$O(r)$	$O(1)$	$O(r)$	$O(r)$	✗
LV [45]	$O(T)^{\diamondsuit}$	$O(1)$	$O(1)$	$O(r)$	$O(1)$	$O(r)$	$O(r)$	✓
LPY1 (SD)	$O(\log N)$	$O(1)$	$O(\log^3 N)$	$O(r)$	$O(\log N)^{\dagger}$	$O(1)$	$O(r \cdot \log N)$	✓
LPY2 (CS)	$O(1)$	$O(1)$	$O(\log N)$	$O(r \cdot \log(N/r))$	$O(1)$	$O(1)$	$O(r \cdot \log(N/r))$	✓
This work	$O(\log N)$	$O(1)$	$O(1)$	$O(r)$	$O(1)$	$O(1)$	$O(r)$	✓

N: max. number of users; r: number of revocations; T: max. number of revocation epochs

$\diamondsuit$ These schemes can be modified to have $O(1)$-size group public keys.

$\dagger$ This complexity is only involved at the first signature of each revocation epoch.

As previously mentioned, among schemes where revocations require no update in unrevoked users' credentials, the new method seems asymptotically optimal. The only dependency on N appears in the group public key size, which is logarithmic and thus quite moderate. At the same time, it retains revocation lists of size $O(r)$ (which is on par with the VLR-based approach [17] but without its verification cost of $O(r)$) as in the SD method of [47]. In comparison with the latter, we also eliminate the $O(\log N)$ multiplicative factor in the revocation cost and the complexity of the signing algorithm in the worst case.

The joining protocol is also much more efficient in our scheme than in [47] as the group manager has to generate only one structure-preserving signature (computing C_v in step 2.a of the protocol is actually cheaper than a single exponentiation in $\mathbb{G}$), instead of $\log(N)$ in the two schemes of [47].

In Appendix C, we give tradeoffs between the strength of the assumption and the efficiency: in these alternative constructions, the assumption is weakened at the expense of group public keys of size $O(\log^2 N)$ or $O(\lambda + \log^2 N)$.

References

1. M. Abe, K. Haralambiev, M. Ohkubo. Signing on Elements in Bilinear Groups for Modular Protocol Design. Cryptology ePrint Archive: Report 2010/133, 2010.
2. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, M. Ohkubo. Structure-Preserving Signatures and Commitments to Group Elements. In *Crypto'10*, *LNCS* 6223, pp. 209–236, 2010.
3. M. Abe, J. Groth, K. Haralambiev, M. Ohkubo. Optimal Structure-Preserving Signatures in Asymmetric Bilinear Groups. In *Crypto'11*, *LNCS* 6841, pp. 649–666, 2011.
4. T. Acar, L. Nguyen. Revocation for Delegatable Anonymous Credentials. In *PKC'11*, *LNCS* 6571, pp. 423–440, 2011.
5. G. Ateniese, J. Camenisch, S. Hohenberger, B. de Medeiros. Practical group signatures without random oracles. Cryptology ePrint Archive: Report 2005/385, 2005.
6. G. Ateniese, J. Camenisch, M. Joye, G. Tsudik. A practical and provably secure coalition-resistant group signature scheme. In *Crypto'00*, *LNCS* 1880, pp. 255–270, 2000.
7. G. Ateniese, D. Song, G. Tsudik. Quasi-Efficient Revocation in Group Signatures. In *Financial Cryptography'02*, *LNCS* 2357, pp. 183–197, 2002.
8. M. Bellare, D. Micciancio, B. Warinschi. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In *Eurocrypt'03*, *LNCS* 2656, pp. 614–629, 2003.
9. M. Bellare, P. Rogaway. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In *1st ACM Conference on Computer and Communications Security*, pp. 62–73, ACM Press, 1993.
10. M. Bellare, H. Shi, C. Zhang. Foundations of group signatures: The case of dynamic groups. In *CT-RSA'05*, *LNCS* 3376, pp. 136–153, 2005.
11. J. Benaloh, M. de Mare. One-Way Accumulators: A Decentralized Alternative to Digital Sinatures. In *Eurocrypt'93*, *LNCS* 4948, pp. 274–285, 1993.
12. D. Boneh, X. Boyen. Short Signatures Without Random Oracles. In *Eurocrypt'04*, *LNCS* 3027, pp. 56–73. Springer-Verlag, 2004.
13. D. Boneh, X. Boyen, E.-J. Goh. Hierarchical identity based encryption with constant size ciphertext. In *Eurocrypt'05*, *LNCS* 3494, pp. 440–456, 2005.
14. D. Boneh, X. Boyen, H. Shacham. Short Group Signatures. In *Crypto'04*, *LNCS* 3152, pp. 41–55. Springer, 2004.
15. D. Boneh and M. Franklin. Identity based encryption from the Weil pairing. *SIAM J. of Computing*, 32(3):586–615, 2003. Extended abstract in Crypto'01, LNCS 2139, pp. 213–229, 2001.
16. D. Boneh, C. Gentry and B. Waters. Collusion-Resistant Broadcast Encryption with Short Ciphertexts and Private Keys. In *Crypto'05*, *LNCS* 3621, pp. 258–275, 2005.
17. D. Boneh, H. Shacham. Group signatures with verifier-local revocation. In *ACM-CCS'04*, pp. 168–177. ACM Press, 2004.
18. X. Boyen, B. Waters. Compact Group Signatures Without Random Oracles. In *Eurocrypt'06*, *LNCS* 4004, pp. 427–444, Springer, 2006.
19. X. Boyen, B. Waters. Full-Domain Subgroup Hiding and Constant-Size Group Signatures. In *PKC'07*, *LNCS* 4450, pp. 1–15, 2007.
20. E. Bresson, J. Stern. Efficient Revocation in Group Signatures. In *PKC'01*, *LNCS* 1992, pp. 190–206, 2001.
21. E. Brickell. An efficient protocol for anonymously providing assurance of the container of the private key. Submission to the Trusted Computing Group. April, 2003.
22. E. Brickell, J. Camenisch, L. Chen. Direct Anonymous Attestation. In *ACM-CCS'04*, pp. 132–145, 2004.
23. J. Camenisch, R. Chaabouni, a. shelat. Efficient Protocols for Set Membership and Range Proofs. In *Asiacrypt'08*, *LNCS* 5350, pp. 234–252, Springer, 2008.
24. J. Camenisch, M. Kohlweiss, C. Soriente. An Accumulator Based on Bilinear Maps and Efficient Revocation for Anonymous Credentials. In *PKC'09*, *LNCS* 5443, pp. 481–500, 2009.

25. J. Camenisch, M. Kohlweiss, C. Soriente. Solving Revocation with Efficient Update of Anonymous Credentials. In *SCN'10*, *LNCS* 6280, pp. 454–471, 2010.
26. J. Camenisch, A. Lysyanskaya. Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In *Crypto'02*, *LNCS* 2442, pp. 61–76, Springer, 2002.
27. D. Catalano, D. Fiore. Concise Vector Commitments and their Applications to Zero-Knowledge Elementary Databases. In Cryptology ePrint Archive: Report 2011/495, 2011.
28. J. Cathalo, B. Libert, M. Yung. Group Encryption: Non-Interactive Realization in the Standard Model. In *Asiacrypt'09*, *LNCS* 5912, pp. 179–196, 2009.
29. D. Chaum, E. van Heyst. Group Signatures. In *Eurocrypt'91*, *LNCS* 547, pp. 257–265, Springer, 1991.
30. C. Delerablée, D. Pointcheval. Dynamic Fully Anonymous Short Group Signatures. In *Vietcrypt'06*, *LNCS* 4341, pp. 193–210, Springer, 2006.
31. Y. Dodis, N. Fazio. Public Key Broadcast Encryption for Stateless Receivers. In *Digital Rights Management (DRM'02)*, *LNCS* 2696, pp. 61–80, 2002.
32. G. Fuchsbauer. Automorphic Signatures in Bilinear Groups and an Application to Round-Optimal Blind Signatures. Cryptology ePrint Archive: Report 2009/320, 2009.
33. C. Gentry, A. Silverberg. Hierarchical ID-based cryptography. In *Asiacrypt'02*, *LNCS* 2501, Springer, 2002.
34. J. Groth. Simulation-Sound NIZK Proofs for a Practical Language and Constant Size Group Signatures. In *Asiacrypt'06*, *LNCS* 4284, pp. 444–459, Springer, 2006.
35. J. Groth. Fully anonymous group signatures without random oracles. In *Asiacrypt 2007*, *LNCS* 4833, pp. 164–180. Springer, 2007.
36. J. Groth, A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Eurocrypt'08*, LNCS 4965, pp. 415–432, 2008.
37. D. Halevy, A. Shamir. The LSD broadcast encryption scheme. In *Crypto'02*, *LNCS* 2442, pp. 47–60, Springer, 2002.
38. J. Horwitz, B. Lynn. Toward hierarchical identity-based encryption. In *Eurocrypt'02*, *LNCS* 2332, Springer, 2002.
39. M. Izabachène, B. Libert, D. Vergnaud. Blockwise P-Signatures and Non-Interactive Anonymous Credentials with Efficient Attributes. 13*th IMA International Conference on Cryptography and Coding (IMACC 2011)*, pp. 431–450, Springer, 2011.
40. A. Kiayias, M. Yung. Secure scalable group signature with dynamic joins and separable authorities. International Journal of Security and Networks (IJSN) Vol. 1, No. 1/2, pp. 24–45, 2006. Earlier version appeared as Cryptology ePrint Archive: Report 2004/076, 2004.
41. A. Kiayias, M. Yung. Group signatures with efficient concurrent join. In *Eurocrypt'05*, *LNCS* 3494, pp. 198–214, 2005.
42. E. Kiltz. Chosen-ciphertext security from tag-based encryption. In *TCC'06*, *LNCS* 3876, pp. 581–600, 2006.
43. S. Kunz-Jacques and D. Pointcheval. About the security of MTI/C0 and MQV. In *SCN'06*, LNCS 4116, pages 156–172, 2006.
44. F. Laguillaumie, P. Paillier, D. Vergnaud. Universally Convertible Directed Signatures. In *Asiacrypt'05*, *LNCS* 3788, pp. 682–701, 2005.
45. B. Libert, D. Vergnaud. Group Signatures with Verifier-Local Revocation and Backward Unlinkability in the Standard Model. In *CANS'09*, *LNCS* 5888, pp. 498-517, 2009.
46. B. Libert and M. Yung. Concise Mercurial Vector Commitments and Independent Zero-Knowledge Sets with Short Proofs. In *TCC'10*, LNCS 5978, pp. 499–517, 2010.
47. B. Libert, T. Peters and M. Yung. Scalable Group Signatures with Revocation. In *Eurocrypt'12*, *LNCS* series, to appear, 2012.
48. T. Malkin, I. Teranishi, Y. Vahlis, M. Yung. Signatures resilient to continual leakage on memory and computation. In *TCC'11*, *LNCS* 6597, pp. 89–106, 2011.
49. T. Nakanishi, H. Fujii, Y. Hira, N. Funabiki. Revocable Group Signature Schemes with Constant Costs for Signing and Verifying. In *PKC'09*, *LNCS* 5443, pp. 463–480, 2009.
50. T. Nakanishi, N. Funabiki. Verifier-Local Revocation Group Signature Schemes with Backward Unlinkability from Bilinear Maps. In *Asiacrypt'05*, *LNCS* 5443, pp. 533–548, 2009.
51. M. Naor, D. Naor, J. Lotspiech. Revocation and Tracing Schemes for Stateless Receivers. In *Crypto'01*, *LNCS* 2139, pp. 41–62, 2001.
52. M. Naor. On Cryptographic Assumptions and Challenges. In *Crypto'03*, *LNCS* 2729, pp. 96–109. Springer-Verlag, 2003.
53. L. Nguyen. Accumulators from Bilinear Pairings and Applications. In *CT-RSA'05*, *LNCS* 3376, pp. 275–292, 2005.

54. L. Nguyen, R. Safavi-Naini. Efficient and Provably Secure Trapdoor-Free Group Signature Schemes from Bilinear Pairings. In *Asiacrypt'04*, *LNCS* 3329, pp. 372–386. Springer-Verlag, 2004.
55. V. Shoup. Lower bounds for discrete logarithms and related problems. In *Eurocrypt'97*, *LNCS* 1233, pp. 256–66, 1997.
56. D. Song. Practical forward secure group signature schemes. In *ACM-CCS'01*, pp. 225–234, 2001.
57. P. Tsang, M.-Ho Au, A. Kapadia, S. Smith. Blacklistable anonymous credentials: blocking misbehaving users without TTPs. In *ACM-CCS'07*, pp. 72–81, 2007.
58. P. Tsang, M.-Ho Au, A. Kapadia, S. Smith. PEREA: towards practical TTP-free revocation in anonymous authentication. In *ACM-CCS'08*, pp. 333–344, 2008.
59. G. Tsudik, S. Xu. Accumulating Composites and Improved Group Signing. In *Asiacrypt'03*, *LNCS* 2894, pp. 269–286, 2003.
60. B. Waters. Efficient identity-based encryption without random oracles. In *Eurocrypt 2005*, *LNCS* 2567. Springer, 2005.
61. S. Zhou, D. Lin. Shorter Verifier-Local Revocation Group Signatures from Bilinear Maps. In *CANS'06*, *LNCS* 4301, pp. 126–143, Springer, 2006.

A Correctness and Security Definitions for Revocable Group Signatures

In the following, as in [40], we say that a public state St is *valid* if it is reachable from $St = (\emptyset, \varepsilon)$ by a Turing machine having oracle access to J_{GM}. Also, a state St' is said to *extend* another state St if it is within reach from St.

As in [40, 41], when we write $\mathsf{cert}_i \leftrightharpoons_{\mathcal{Y}} \mathsf{sec}_i$, it means that there exist coin tosses ϖ for J_{GM} and J_{user} such that, for some valid public state St', the execution of $[\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y}), \mathsf{J}_{\mathsf{GM}}(\lambda, St', \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})](\varpi)$ provides $\mathsf{J}_{\mathsf{user}}$ with $\langle i, \mathsf{sec}_i, \mathsf{cert}_i \rangle$.

Correctness. A R-GS scheme is correct if the following conditions are all satisfied:

1. In a valid state St, it always holds that $|St_{users}| = |St_{trans}|$ and two distinct entries of St_{trans} always contain certificates with distinct tag.
2. If the protocol $[\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y}), \mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})]$ is run by two honest parties and $\langle i, \mathsf{cert}_i, \mathsf{sec}_i \rangle$ is obtained by $\mathsf{J}_{\mathsf{user}}$, then it holds that $\mathsf{cert}_i \leftrightharpoons_{\mathcal{Y}} \mathsf{sec}_i$.
3. For each revocation epoch t and any $\langle i, \mathsf{cert}_i, \mathsf{sec}_i \rangle$ such that $\mathsf{cert}_i \leftrightharpoons_{\mathcal{Y}} \mathsf{sec}_i$, satisfying condition 2, if $i \notin \mathcal{R}_t$, it always holds that $\mathsf{Verify}\big(\mathsf{Sign}(\mathcal{Y}, t, RL_t, \mathsf{cert}_i, \mathsf{sec}_i, M), M, t, RL_t, \mathcal{Y}\big) = 1$.
4. For any outcome $\langle i, \mathsf{cert}_i, \mathsf{sec}_i \rangle$ of the interaction $[\mathsf{J}_{\mathsf{user}}(.,.), \mathsf{J}_{\mathsf{GM}}(., St, ., .)]$ for some valid state St, any revocation epoch t such that $i \notin \mathcal{R}_t$, if $\sigma = \mathsf{Sign}(\mathcal{Y}, t, RL_t, \mathsf{cert}_i, \mathsf{sec}_i, M)$, then

$$\mathsf{Open}(M, t, RL_t, \sigma, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St') = i.$$

Security Model. As in [40], we formalize security properties via experiments where the adversary interacts with a stateful interface $\mathcal{I}$ that maintains the following variables:

- $\mathsf{state}_{\mathcal{I}}$: is a data structure representing the state of the interface as the adversary invokes the various oracles. It is initialized as $\mathsf{state}_{\mathcal{I}} = (St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, \mathcal{S}_{\mathsf{OA}}) \leftarrow \mathsf{Setup}(\lambda, N)$. It includes the (initially empty) set St_{users} of group members and a dynamically growing database St_{trans} storing the transcripts of previously executed join protocols. Finally, $\mathsf{state}_{\mathcal{I}}$ includes a counter t (which is initialized to 0) indicating the number of user revocation queries so far.
- $n = |St_{users}| < N$ denotes the current cardinality of the group.
- Sigs: is a database of signatures created by the signing oracle. Each entry consists of a triple (i, t, M, σ) indicating that message M was signed by user i at epoch t.

- U^a: is the set of users that were introduced by the adversary in the system in an execution of the join protocol.
- U^b: is the set of honest users that the adversary, acting as a dishonest group manager, introduced in the system. For these users, the adversary obtains the transcript of the join protocol but not the user's membership secret.

When mounting attacks, adversaries will be granted access to the following oracles.

- Q_{pub}, Q_{keyGM} and Q_{keyOA}: when these oracles are invoked, the interface looks up $\mathsf{state}_{\mathcal{I}}$ and returns the group public key $\mathcal{Y}$, the GM's private key $\mathcal{S}_{\mathsf{GM}}$ and the opening authority's private key $\mathcal{S}_{\mathsf{OA}}$ respectively.
- $Q_{\mathsf{a\text{-}join}}$: allows the adversary to introduce users under his control in the group. On behalf of the GM, the interface runs J_{GM} in interaction with the $\mathsf{J}_{\mathsf{user}}$-executing adversary who plays the role of the prospective user in the join protocol. If this protocol successfully ends, the interface increments N, updates St by inserting the new user n in both sets St_{users} and U^a. It also sets $St_{\mathsf{trans}} := St_{\mathsf{trans}}||\langle n, \mathsf{transcript}_n\rangle$.
- $Q_{\mathsf{b\text{-}join}}$: allows the adversary, acting as a corrupted group manager, to introduce new honest group members of his choice. The interface triggers an execution of $[\mathsf{J}_{\mathsf{user}}, \mathsf{J}_{\mathsf{GM}}]$ and runs $\mathsf{J}_{\mathsf{user}}$ in interaction with the adversary who runs J_{GM}. If the protocol successfully completes, the interface increments n, adds user n to St_{users} and U^b and sets $St_{\mathsf{trans}} := St_{\mathsf{trans}}||\langle n, \mathsf{transcript}_n\rangle$. It stores the membership certificate cert_n and the membership secret sec_n in a *private* part of $\mathsf{state}_{\mathcal{I}}$.
- Q_{sig}: given a message M, an index i, the interface checks if the private area of $\mathsf{state}_{\mathcal{I}}$ contains a certificate cert_i and a membership secret sec_i such that $i \notin \mathcal{R}_t$, where t is the current revocation epoch. If no such elements $(\mathsf{cert}_i, \mathsf{sec}_i)$ exist or if $i \notin U^b$, the interface returns $\perp$. Otherwise, it outputs a signature σ on behalf of user i for epoch t and also sets $\mathsf{Sigs} \leftarrow \mathsf{Sigs}||(i, t, M, \sigma)$.
- Q_{open}: when this oracle is invoked on input of a valid pair (M, σ) for some revocation epoch t, the interface runs algorithm Open using the current state St. When S is a set of triples of the form (M, σ, t), $Q_{\mathsf{open}}^{\neg S}$ denotes a restricted oracle that only applies the opening algorithm to triples (M, σ, t) which are not in S.
- Q_{read} and Q_{write}: are used by the adversary to read and write the content of $\mathsf{state}_{\mathcal{I}}$. Namely, at each invocation, Q_{read} outputs the whole $\mathsf{state}_{\mathcal{I}}$ but the public/private keys and the private part of $\mathsf{state}_{\mathcal{I}}$ where membership secrets are stored after $Q_{\mathsf{b\text{-}join}}$-queries. By using Q_{write}, the adversary can modify $\mathsf{state}_{\mathcal{I}}$ at will as long as it does not remove or alter elements of St_{users}, St_{trans} or invalidate the public state St: for example, the adversary is allowed to create dummy users as long as it does not re-use already existing certificate tags.
- Q_{revoke}: is a revocation oracle. Given an index i such that $i \in St_{users}$, the interface checks if i appears in the appropriate user set (namely, U^a or U^b depending on the considered security notion) and if the database St_{trans} contains a record $\langle i, \mathsf{transcript}_i\rangle$ such that $i \notin \mathcal{R}_t$, where t is the current revocation epoch. If not, it returns $\perp$. Otherwise, it increments t, adds i to $\mathcal{R}_t$ and generates an updated revocation list RL_t which is made available to the adversary. For simplicity, we assumed that the adversary only revokes one user per query to Q_{revoke} but the model easily extends to allow multiple revocations at once.

The Kiayias-Yung model considers properties called security against *misidentification attacks*, *framing attacks* and *anonymity*.

In a misidentification attack, the adversary can corrupt the opening authority using the Q_{keyOA}

oracle. Moreover, he can also introduce malicious users in the group via $Q_{\mathsf{a\text{-}join}}$-queries and revoke users at any time using Q_{revoke}. His purpose is to come up with a signature $\sigma^\star$ that verifies w.r.t. $RL_{t^\star}$, where $t^\star$ denotes the current revocation epoch (*i.e.*, the number of Q_{revoke}-queries). He is deemed successful if the produced signature $\sigma^\star$ does not open to any unrevoked adversarially-controlled.

Definition 6. *A R-GS scheme is secure against misidentification attacks if, for any PPT adversary $\mathcal{A}$ involved in the experiment hereafter, we have* $\mathbf{Adv}_{\mathcal{A}}^{\text{mis-id}}(\lambda) = \Pr[\mathbf{Expt}_{\mathcal{A}}^{\text{mis-id}}(\lambda) = 1] \in \mathsf{negl}(\lambda)$.

Experiment $\mathbf{Expt}_{\mathcal{A}}^{\text{mis-id}}(\lambda)$
$\mathsf{state}_{\mathcal{I}} = (St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, \mathcal{S}_{\mathsf{OA}}) \leftarrow \mathsf{Setup}(\lambda, N)$;
$(M^\star, \sigma^\star) \leftarrow \mathcal{A}(Q_{\mathsf{pub}}, Q_{\mathsf{a\text{-}join}}, Q_{\mathsf{revoke}}, Q_{\mathsf{read}}, Q_{\mathsf{keyOA}})$;
If $\mathsf{Verify}(\sigma^\star, M, t^\star, RL_{t^\star}, \mathcal{Y}) = 0$ *return* 0;
$i = \mathsf{Open}(M^\star, t^\star, RL_{t^\star}, \sigma^\star, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St')$;
If $(i \notin U^a \backslash \mathcal{R}_{t^\star})$ *return* 1;
Return 0;

This definition extends the usual definition [40] in that $\mathcal{A}$ also wins if his forgery $\sigma^\star$ verifies w.r.t. $RL_{t^\star}$ but opens to an adversarially-controlled user that *was* revoked during the revocation epoch $t^\star$.

Framing attacks consider the situation where the entire system, including the group manager and the opening authority, is colluding against some honest user. The adversary can corrupt the group manager as well as the opening authority (via oracles Q_{keyGM} and Q_{keyOA}, respectively). He is also allowed to introduce honest group members (via $Q_{\mathsf{b\text{-}join}}$-queries), observe the system while these users sign messages and create dummy users using Q_{write}. In addition, before the possible corruption of the group manager, the adversary can revoke group members at any time by invoking the Q_{revoke} oracle. As a potentially corrupted group manager, $\mathcal{A}$ is allowed to come up with his own revocation list $RL_{t^\star}$ at the end of the game. We assume that anyone can publicly verify that $RL_{t^\star}$ is correctly formed (*i.e.*, that it could be a legitimate output of Revoke) so that the adversary does not come up with an ill-formed revocation list. For consistency, if $\mathcal{A}$ chooses not to corrupt the GM, the produced revocation list $RL_{t^\star}$ must be the one determined by the history of Q_{revoke}-queries. The adversary eventually aims at framing an honest group member.

Definition 7. *A R-GS scheme is secure against framing attacks if, for any PPT adversary $\mathcal{A}$, it holds that* $\mathbf{Adv}_{\mathcal{A}}^{\text{fra}}(\lambda) = \Pr[\mathbf{Expt}_{\mathcal{A}}^{\text{fra}}(\lambda) = 1] \in \mathsf{negl}(\lambda)$.

Experiment $\mathbf{Expt}_{\mathcal{A}}^{\text{fra}}(\lambda)$
$\mathsf{state}_{\mathcal{I}} = (St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, \mathcal{S}_{\mathsf{OA}}) \leftarrow \mathsf{Setup}(\lambda, N)$;
$(M^\star, \sigma^\star, t^\star, RL_{t^\star}) \leftarrow \mathcal{A}(Q_{\mathsf{pub}}, Q_{\mathsf{keyGM}}, Q_{\mathsf{keyOA}}, Q_{\mathsf{b\text{-}join}}, Q_{\mathsf{revoke}}, Q_{\mathsf{sig}}, Q_{\mathsf{read}}, Q_{\mathsf{write}})$;
If $\mathsf{Verify}(\sigma^\star, M^\star, t^\star, RL_{t^\star}, \mathcal{Y}) = 0$ *then return* 0;
$i = \mathsf{Open}(M^\star, t^\star, RL_{t^\star}, \sigma^\star, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St')$;
If $i \notin U^b$ *return* 0;
If $\big(\bigwedge_{j \in U^b \text{ s.t. } j=i} (j, t^\star, M^\star, *) \notin \mathsf{Sigs}\big)$ *then return* 1;
Return 0;

The notion of anonymity is formalized by means of a game involving a two-stage adversary. In the following, we assume that, from a given valid membership certificate/secret pair $(\mathsf{cert}, \mathsf{sec})$ and a given revocation list RL_t, it is easy to decide if $(\mathsf{cert}, \mathsf{sec})$ belongs to a revoked user for RL_t. More precisely, there must exist an efficient algorithm $\mathsf{IsRevoked}$ that takes as input $(\mathsf{sec}, \mathsf{cert}, RL_t)$

and returns 1 if the pair $(\mathsf{sec}, \mathsf{cert})$ is not the key material of an unrevoked user for RL_t (such an algorithm obviously exists in our construction).

The first stage of the game is called play stage and allows the adversary $\mathcal{A}$ to modify $\mathsf{state}_{\mathcal{I}}$ via Q_{write}-queries and to open arbitrary signatures by probing Q_{open}. When the play stage ends, the adversary $\mathcal{A}$ chooses a message-period pair $(M^\star, t^\star)$, a revocation list $RL_{t^\star}$ as well as two pairs $(\mathsf{sec}_0^\star, \mathsf{cert}_0^\star)$, $(\mathsf{sec}_1^\star, \mathsf{cert}_1^\star)$, consisting of a valid membership certificate and a corresponding membership secret satisfying $\mathsf{IsRevoked}(\mathsf{sec}_b^\star, \mathsf{cert}_b^\star, RL_{t^\star}) = 0$ for each $b \in \{0,1\}$. Then, the challenger flips a coin $d \xleftarrow{R} \{0,1\}$ and computes a challenge signature $\sigma^\star$ using $(\mathsf{sec}_d^\star, \mathsf{cert}_d^\star)$. The adversary is given $\sigma^\star$ with the task of eventually guessing the bit $d \in \{0,1\}$. Before doing so, he is allowed further oracle queries throughout the second stage, called guess stage, but is restricted not to query Q_{open} for $(M^\star, \sigma^\star, t^\star)$.

Definition 8. *A R-GS scheme is fully anonymous if* $\mathbf{Adv}^{\mathrm{anon}}(\mathcal{A}) := |\Pr[\mathbf{Expt}_{\mathcal{A}}^{\mathrm{anon}}(\lambda) = 1] - 1/2|$ *is negligible for any PPT adversary* $\mathcal{A}$ *involved in the following experiment:*

Experiment $\mathbf{Expt}_{\mathcal{A}}^{\mathrm{anon}}(\lambda)$
$\mathsf{state}_{\mathcal{I}} = (St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, \mathcal{S}_{\mathsf{OA}}) \leftarrow \mathsf{Setup}(\lambda)$;
$\big(aux, M^\star, t^\star, RL_{t^\star}, (\mathsf{sec}_0^\star, \mathsf{cert}_0^\star), (\mathsf{sec}_1^\star, \mathsf{cert}_1^\star)\big)$
$\quad \leftarrow \mathcal{A}(\mathsf{play} : Q_{\mathsf{pub}}, Q_{\mathsf{keyGM}}, Q_{\mathsf{revoke}}, Q_{\mathsf{open}}, Q_{\mathsf{read}}, Q_{\mathsf{write}})$;
If $\neg(\mathsf{cert}_b \leftrightharpoons_{\mathcal{Y}} \mathsf{sec}_b)$ *or* $\mathsf{IsRevoked}(\mathsf{sec}_b^\star, \mathsf{cert}_b^\star, RL_{t^\star}) = 1$ *for* $b \in \{0,1\}$
$\quad$ *or if* $\mathsf{cert}_0^\star = \mathsf{cert}_1^\star$ *return* 0;
$d \xleftarrow{R} \{0,1\}$; $\sigma^\star \leftarrow \mathsf{Sign}(\mathcal{Y}, t^\star, \mathsf{cert}_d^\star, \mathsf{sec}_d^\star, M^\star)$;
$d' \leftarrow \mathcal{A}(\mathsf{guess} : \sigma^\star, aux, Q_{\mathsf{pub}}, Q_{\mathsf{keyGM}}, Q_{\mathsf{open}}^{\neg\{(M^\star, \sigma^\star, t^\star)\}}, Q_{\mathsf{read}}, Q_{\mathsf{write}})$;
If $d' = d$ *then return* 1;
Return 0;

B Security Proofs

B.1 Security Against Misidentification Attacks

Theorem 2 (Misidentification). *The scheme is secure against misidentification attacks assuming that the q-SFP and the ℓ-FlexDHE problems are both hard for $q = \max(q_a, q_r^2)$ and $\ell = \log N$, where q_a and q_r denote the maximal numbers of $Q_{\mathsf{a\text{-}join}}$ queries and Q_{revoke} queries, respectively, and N is the maximal number of group members.*

Proof. Towards a contradiction, let us assume that the adversary $\mathcal{A}$ outputs a non-trivial signature that does not open to an unrevoked adversarially-controlled group member.

Let $\sigma^\star = \big(\mathsf{VK}^\star, \Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star, \Upsilon_4^\star, \Upsilon_5^\star, \Omega^\star, \mathbf{com}^\star, \mathbf{\Pi}^\star, \sigma_{ots}^\star\big)$ denote $\mathcal{A}$'s forgery and parse $\mathbf{com}^\star$ as

$$\mathbf{com}^\star = \big(com^\star_{C_{v_i}}, com^\star_X, \{com^\star_{R_{l,\tau}}\}_{\tau=2}^5, com^\star_{W_{\phi_l}}, com^\star_{W_{\psi_l}}, com^\star_{\Gamma_l}, \{com^\star_{\Psi_{l,\tau}}\}_{\tau\in\{0,1,2\ell\}}, \{com^\star_{\Theta'_{l,j}}\}_{j\in\{1,2,5\}}, \{com^\star_{\theta'_{l,j}}\}_{j\in\{1,2,5\}}, \{com^\star_{\chi_j}\}_{j=1}^3, com^\star_{\sigma_{\mathsf{VK}}}\big)$$

We thus have $\mathsf{Open}(M^\star, t^\star, RL_{t^\star}, \sigma^\star, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St) \notin U^a \backslash \mathcal{R}_{t^\star}$, where U^a denotes the set of adversarially-controlled users. Depending on the contents of extractable commitments $com^\star_X$, $com^\star_{C_{v_i}}$, $\{com^\star_{R_{l,\tau}}\}_{\tau=2}^5$, $\{com^\star_{\Psi_{l,\tau}}\}_{i\in\{0,1,2\ell\}}$, $com^\star_{W_{\phi_l}}$, $com^\star_{W_{\psi_l}}$, $com^\star_{\Gamma_l}$, we distinguish the following cases:

- **Type I forgeries** are those for which $\{com^\star_{R_{l,\tau}}\}_{\tau=2}^5$ contain group elements $(R^\star_{l,2}, \ldots, R^\star_{l,5})$ such that $(g^{t^\star}, R^\star_{l,2}, \ldots, R^\star_{l,5})$ was never signed when the latest revocation list $RL_{t^\star}$ was generated.

- **Type II forgeries** are such that $\{com^{\star}_{R_{l,\tau}}\}_{\tau=2}^{5}$ contain group elements $(R^{\star}_{l,2}, \ldots, R^{\star}_{l,5})$ for which the message $(g^{t^{\star}}, R^{\star}_{l,2}, \ldots, R^{\star}_{l,5})$ was signed when the latest revocation list $RL_{t^{\star}}$ was publicized at epoch $t^{\star}$. At the same time, Open uncovers a user's tag $X^{\star}$ for which one of the following two situations occurs:
  a. The pair $(X^{\star}, C^{\star}_{v_i})$ was not signed using $sk^{(0)}_{\mathsf{AHO}}$.
  b. $(X^{\star}, C^{\star}_{v_i})$ was signed when answering some $Q_{\mathsf{a\text{-}join}}$-query. However, $C^{\star}_{v_i}$ encodes the path $(I^{\star}_1, \ldots, I^{\star}_{\ell})$ of a leaf $v^{\star}_i$ assigned to a revoked user $i^{\star}$ even though the forgery $\sigma^{\star}$ provides convincing evidence that the committed values $C^{\star}_{v_i}$, $(R^{\star}_{l,2}, R^{\star}_{l,3}, R^{\star}_{l,4}, R^{\star}_{l,5})$, $(\Psi^{\star}_{l,0}, \Psi^{\star}_{l,1}, \Psi^{\star}_{l,2\ell})$ and $(\Gamma^{\star}_{l}, W^{\star}_{\phi_l}, W^{\star}_{\psi_l})$ satisfy the relations

$$e(R^{\star}_{l,2}, C^{\star}_{v_i}) = e(R^{\star}_{l,3}, g_{\ell}) \cdot e(g, W^{\star}_{\phi_l}), \tag{11}$$

and

$$e(R^{\star}_{l,4}, C^{\star}_{v_i}) = e(\Psi^{\star}_{l,1}, g_{\ell}) \cdot e(g, W^{\star}_{\psi_l}) \tag{12}$$

$$e(\Psi^{\star}_{l,0}/R^{\star}_{l,5}, \Gamma^{\star}_{l}) = e(g, g) \tag{13}$$

$$e(\Psi^{\star}_{l,1}, g) = e(g_1, \Psi^{\star}_{l,0}) \tag{14}$$

$$e(\Psi^{\star}_{l,2\ell}, g) = e(g_{2\ell}, \Psi^{\star}_{l,0}). \tag{15}$$

It is immediate that Type I and Type II.a forgeries imply a forger against the AHO signature scheme and the proof is omitted.

Lemma 1 demonstrates that a Type II.b forgery necessarily contradicts the ℓ-FlexDHE assumption. This completes the proof since $\sigma^{\star}$ cannot constitute a successful misidentification attack without being a Type I or a Type II forgery. □

Lemma 1. *The advantage of any Type II.b forger $\mathcal{A}$ is at most*

$$\mathbf{Adv}^{\text{mis-id-II.b}}_{\mathcal{A}}(\lambda) \leq \mathbf{Adv}^{\ell\text{-FlexDHE}}(\lambda)$$

where $\ell = \log N$ and N denotes the maximal number of users.

Proof. The reduction $\mathcal{B}$ takes as input a ℓ-FlexDHE instance $(g, g_1, \ldots, g_{\ell}, g_{\ell+2}, \ldots, g_{2\ell}) \in \mathbb{G}^{2\ell}$. To generate the group public key $\mathcal{Y}$ it follows exactly the specification of the Setup algorithm with the difference that, instead of computing ck as per step 3 of the algorithm, it defines $ck = (g_1, \ldots, g_{\ell}, g_{\ell+2}, \ldots, g_{2\ell}) \in \mathbb{G}^{2\ell-1}$ using its input and gives $\mathcal{Y} := \big(g, pk^{(0)}_{\mathsf{AHO}}, pk^{(1)}_{\mathsf{AHO}}, ck, \mathbf{f}, \vec{\varphi}, (U, V), \Sigma\big)$ to the Type II.b forger $\mathcal{A}$.

Throughout the game, the adversary can adaptively invoke the Q_{pub}, $Q_{\mathsf{a\text{-}join}}$, Q_{revoke}, Q_{read}, and Q_{keyOA} oracles. Since $\mathcal{B}$ knows $\mathcal{S}_{\mathsf{GM}} = (sk^{(0)}_{\mathsf{AHO}}, sk^{(1)}_{\mathsf{AHO}})$ and $\mathcal{S}_{\mathsf{OA}} = (\beta_1, \beta_2)$, it can faithfully answer all adversarial queries. The game ends with the adversary outputting a forgery $\sigma^{\star}$ for which the committed variables $C^{\star}_{v_i}$, $(R^{\star}_{l,2}, R^{\star}_{l,3}, R^{\star}_{l,4}, R^{\star}_{l,5})$, $(\Psi^{\star}_{l,0}, \Psi^{\star}_{l,1}, \Psi^{\star}_{l,2\ell})$ and $(\Gamma^{\star}_{l}, W^{\star}_{\phi_l}, W^{\star}_{\psi_l})$ satisfy relations (11)-(15) although $\sigma^{\star}$ opens to some user $i^{\star} \in U^a \cap \mathcal{R}_{t^{\star}}$.

Note that $(R^{\star}_{l,1}, R^{\star}_{l,2}, R^{\star}_{l,3}, R^{\star}_{l,4}, R^{\star}_{l,5})$ must be of the form

$$(R^{\star}_{l,1}, R^{\star}_{l,2}, R^{\star}_{l,3}, R^{\star}_{l,4}, R^{\star}_{l,5}) = \Big(g^{t^{\star}},\ g_{\phi_l},\ g_1^{\mathsf{ID}(x^{\star}_{k_l})},\ g_{\psi_l},\ g^{\mathsf{ID}(x^{\star}_{k_l})}\Big), \tag{16}$$

for some $\phi_l, \psi_l \in \{1, \ldots, \ell\}$ and some $\mathsf{ID}(x^{\star}_{k_l}), \mathsf{ID}(x^{\star}_{u_l}) \in \{1, \ldots, 2N-1\}$ that $\mathcal{B}$ knows for having chosen them itself at the latest Q_{revoke}-query. By hypothesis, $\sigma^{\star}$ contains a committed pair $(X^{\star}, C^{\star}_{v_i})$

that was signed by $\mathcal{B}$ at some $Q_{\mathsf{a\text{-}join}}$-query. Then, $\mathcal{B}$ recalls $(I_1^\star, \ldots, I_\ell^\star)$ such that $C_{v_i}^\star = \prod_{\kappa=1}^{\ell} g_{\ell+1-\kappa}^{I_\kappa^\star}$ from its interaction with $\mathcal{A}$ at that $Q_{\mathsf{a\text{-}join}}$-query. Since $i^\star \in U^a \cap \mathcal{R}_{t^\star}$, it must hold that either:

- $I_{\phi_l}^\star \neq \mathsf{ID}(x_{k_l}^\star)$: In this case, relations (16) and (11) imply that

$$e(g_{\phi_l}, C_{v_i}^\star) = e(g_1, g_\ell)^{\mathsf{ID}(x_{k_l}^\star)} \cdot e(g, W_{\phi_l}^\star) \tag{17}$$

for values $\phi_l \in \{1, \ldots, \ell\}$ and $\mathsf{ID}(x_{k_l})^\star \in \{1, \ldots, 2N-1\}$ that are available to $\mathcal{B}$. Since it also knows $(I_1^\star, \ldots, I_\ell^\star)$ such that $C_{v_i}^\star = \prod_{\kappa=1}^{\ell} g_{\ell+1-\kappa}^{I_\kappa^\star}$, it can compute $W' = \prod_{\kappa=1,\ \kappa\neq\phi_l}^{\ell} g_{\ell+1-\kappa+\phi_l}^{I_\kappa^\star}$ which satisfies

$$e(g_{\phi_l}, C_{v_i}^\star) = e(g_1, g_\ell)^{I_{\phi_l}^\star} \cdot e(g, W'). \tag{18}$$

By combining (17) and (18), we find that $g_{\ell+1} = \left(W_{\phi_l}^\star / W'\right)^{1/(I_{\phi_l}^\star - \mathsf{ID}(x_{k_l}^\star))}$ is computable by $\mathcal{B}$ and it solves an instance the ℓ-DHE problem (which is not easier than ℓ-FlexDHE).

- $I_{\psi_l}^\star = \mathsf{ID}(x_{u_l}^\star)$: In this situation, if we define $\varrho = \log_{g_1}(\Psi_{l,1}^\star)$, relations (16) and (12)-(15) imply that

$$e(g_{\psi_l}, C_{v_i}^\star) = e(g_1, g_\ell)^{\varrho} \cdot e(g, W_{\psi_l}^\star) \tag{19}$$

$$g^{\varrho - I_{\psi_l}^\star} \neq 1_{\mathbb{G}} \tag{20}$$

$$\Psi_{l,0}^\star = g^{\varrho} \tag{21}$$

$$\Psi_{l,2\ell}^\star = g_{2\ell}^{\varrho} \tag{22}$$

Also, similarly to the previous case, $\mathcal{B}$ can compute $W' = \prod_{\kappa=1,\ \kappa\neq\psi_l}^{\ell} g_{\ell+1-\kappa+\psi_l}^{I_\kappa^\star}$ such that

$$e(g_{\psi_l}, C_{v_i}^\star) = e(g_1, g_\ell)^{I_{\psi_l}^\star} \cdot e(g, W'). \tag{23}$$

If we divide (19) by (23), we obtain the equality $e(g_1, g_\ell)^{\varrho - I_{\psi_l}^\star} = e(g, W'/W_{\phi_l}^\star)$, so that $W'/W_{\phi_l}^\star = g_{\ell+1}^{\varrho - I_{\psi_l}^\star}$. The triple

$$\left(\Psi_{l,0}^\star \cdot g^{-I_{\psi_l}^\star},\ W'/W_{\phi_l}^\star,\ \Psi_{l,2\ell}^\star \cdot g_{2\ell}^{-I_{\psi_l}^\star}\right) = \left(g^{\varrho - I_{\psi_l}^\star},\ g_{\ell+1}^{\varrho - I_{\psi_l}^\star},\ g_{2\ell}^{\varrho - I_{\psi_l}^\star}\right)$$

thus forms a non-trivial solution to the ℓ-FlexDHE problem.

In either case, we observe that $\mathcal{B}$ solves either the given ℓ-FlexDHE instance or the potentially harder ℓ-DHE problem. □

B.2 Security Against Framing Attacks

The security against framing attacks relies on the SDH assumption and the security of the one-time signature.

Theorem 3 (Non-frameability). *The scheme is secure against framing attacks assuming that: (i) the q_b-SDH assumption holds in $\mathbb{G}$, where q_b is the maximal number of $Q_{\mathsf{b\text{-}join}}$-queries; (ii) Σ is a strongly unforgeable one-time signature.*

Proof. As in [35], we consider two kinds of framing attacks that can be possibly mounted by a non-frameability adversary $\mathcal{A}$.

- **Type I attacks**: $\mathcal{A}$ generates a forgery $\sigma^\star = \big(\mathsf{VK}^\star, \Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star, \Upsilon_4^\star, \Upsilon_5^\star, \Omega^\star, \mathbf{com}^\star, \mathbf{\Pi}^\star, \sigma_{ots}^\star\big)$ for which the one-time verification key $\mathsf{VK}^\star$ was used by some honest group member $i \in U^b$ when answering a Q_{sig}-query.
- **Type II attacks**: $\mathcal{A}$ outputs a forgery $\sigma^\star = \big(\mathsf{VK}^\star, \Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star, \Upsilon_4^\star, \Upsilon_5^\star, \Omega^\star, \mathbf{com}^\star, \mathbf{\Pi}^\star, \sigma_{ots}^\star\big)$ for which the one-time verification key $\mathsf{VK}^\star$ was never used by Q_{sig} to answer a signing query on behalf of an honest user $i \in U^b$.

Type I attacks clearly defeat the security of the one-time signature. Lemma 2 shows that a Type II forgery would contradict the Strong Diffie-Hellman assumption. □

Lemma 2. *The scheme is secure against framing attacks of Type II if the q_s-SDH problem is hard. More precisely, the advantage of any adversary after q_s Q_{sig}-queries and q_b $Q_{\mathsf{b\text{-}join}}$-queries is at most* $\mathbf{Adv}^{\text{fra-II}}(\lambda) \leq q_b \cdot \mathbf{Adv}^{\text{q}_\text{s}\text{-SDH}}(\lambda)$.

Proof. Let us assume that a PPT adversary $\mathcal{A}$ comes up with a forgery $(M^\star, \sigma^\star)$ that opens to some honest user $i \in U^b$ who did not issue a signature containing the verification key $\mathsf{VK}^\star$. The same proof as in [35] shows that the Strong Diffie-Hellman assumption can be broken.

Given a q-SDH instance $(\tilde{g}, \tilde{g}^a, \ldots, \tilde{g}^{(a^{q_s})}) \in \mathbb{G}^{q_s+1}$, the reduction $\mathcal{B}$ generates a set of q_s one-time signature keys pairs $(\mathsf{SK}_i, \mathsf{VK}_i) \leftarrow \mathcal{G}(\lambda)$ for $i = 1$ to q_s. Then, using the Boneh-Boyen techniques (see [12][Lemma 3.2]) it builds a generator g and a randomly distributed public value $X^\dagger = g^a$ – which implicitly defines $x^\dagger = \log_g(X^\dagger) = a$ – such that it knows $\{(g^{1/(a+\mathsf{VK}_i)}, \mathsf{VK}_i)\}_{i=1}^{q_s}$.

Next, using the newly generated g, $\mathcal{B}$ generates key pairs $\{(sk_{\mathsf{AHO}}^{(b)}, pk_{\mathsf{AHO}}^{(b)})\}_{b=0,1}$ for the AHO signature (note that group elements of $\{pk_{\mathsf{AHO}}^{(b)}\}_{b=0,1}$ are computed as powers of g) and uses $pk_{\mathsf{AHO}}^{(0)}, pk_{\mathsf{AHO}}^{(1)}$ to form the group public key

$$\mathcal{Y} := \Big(g,\ pk_{\mathsf{AHO}}^{(0)},\ pk_{\mathsf{AHO}}^{(1)},\ ck,\ \mathbf{f},\ \vec{\varphi},\ (U,V),\ \Sigma\Big).$$

The underlying Groth-Sahai CRS $\mathbf{f} = (\vec{f_1}, \vec{f_2}, \vec{f_3})$ is generated for the perfect soundness setting, *i.e.*, with $\vec{f_1} = (f_1 = g^{\beta_1}, 1, g)$, $\vec{f_2} = (1, f_2 = g^{\beta_2}, g)$ and $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$, where $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$.

If the adversary $\mathcal{A}$ decides to corrupt the group manager or the opening authority during the game, $\mathcal{B}$ can reveal $\mathcal{S}_{\mathsf{GM}} = (sk_{\mathsf{AHO}}^{(0)}, sk_{\mathsf{AHO}}^{(1)})$ and $\mathcal{S}_{\mathsf{OA}} = (\beta_1, \beta_2) = (\log_g(f_1), \log_g(f_2))$. At the outset of the game, $\mathcal{B}$ picks a random $j^\star \xleftarrow{R} \{1, \ldots, q_b\}$ and interacts with the Type II forger $\mathcal{A}$ as follows.

- Q_{keyGM}-queries: if $\mathcal{A}$ decides to corrupt the group manager, $\mathcal{B}$ surrenders $\mathcal{S}_{\mathsf{GM}} = (sk_{\mathsf{AHO}}^{(0)}, sk_{\mathsf{AHO}}^{(1)})$.
- $Q_{\mathsf{b\text{-}join}}$-queries: when $\mathcal{A}$, acting as a corrupted group manager, decides to introduce a new honest user i in the group, $\mathcal{B}$ starts interacting with $\mathcal{A}$ in an execution of Join and runs $\mathsf{J}_{\mathsf{user}}$ on behalf of the honest user. The actions taken by $\mathcal{B}$ depend on the index $j \in \{1, \ldots, q_b\}$ of the $Q_{\mathsf{b\text{-}join}}$-query.
  - If $j \neq j^\star$, $\mathcal{B}$ follows exactly the specification of $\mathsf{J}_{\mathsf{user}}$.
  - If $j = j^\star$, $\mathcal{B}$ sends the value $X^\dagger$ to J_{GM} at step 1 of Join. User $j^\star$'s membership secret is implicitly defined to be the unknown exponent $\mathsf{sec}_{j^\star} = a$ of the q-SDH instance. In steps 2-5 of the join protocol, $\mathcal{B}$ proceeds like the real $\mathsf{J}_{\mathsf{user}}$ algorithm . When Join terminates, $\mathcal{B}$ obtains a membership certificate $\mathsf{cert}_{j^\star} = \big(\mathsf{ID}(v^\star), X^\dagger, C_{v^\star}, \sigma_{v^\star}\big)$.

- Q_{pub}-queries: can be treated as in the real game, by having the simulator return $\mathcal{Y}$.
- Q_{sig}-queries: when the adversary $\mathcal{A}$ asks user $i \in U^b$ to sign a message M, $\mathcal{B}$ can answer the query by running the real signature generation algorithm if $i \neq j^\star$. Otherwise (namely, if $i = j^\star$), $\mathcal{B}$ uses the next available pair $\{(g^{1/(a+\mathsf{VK}_i)}, \mathsf{VK}_i)\}_{i=1}^{q_s}$ to define $\sigma_{\mathsf{VK}_i} = g^{1/(a+\mathsf{VK}_i)}$. It also recalls user $j^\star$'s membership certificate $\mathsf{cert}_{j^\star} = \big(\mathsf{ID}(v^\star), X^\dagger, C_{v^\star}, \sigma_{v^\star}\big)$ that it obtained from the J_{GM}-executing adversary at the $j^\star$-th $Q_{\mathsf{b\text{-}join}}$-query. Using σ_{VK_i} and $\mathsf{cert}_{j^\star}$, it can easily generate all signature components and sign them using the one-time private key SK_i.

Finally, $\mathcal{A}$ outputs a signature $\sigma^\star = \big(\mathsf{VK}^\star, \Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star, \Upsilon_4^\star, \Upsilon_5^\star, \Omega^\star, \mathbf{com}^\star, \mathbf{\Pi}^\star, \sigma_{ots}^\star\big)$, for some message $M^\star$, that opens to some user $i^\star \in U^b$ who did not sign $M^\star$. At this point, $\mathcal{B}$ halts and reports failure if it turns out that $X^\dagger \neq \Upsilon_3^\star \cdot \Upsilon_1^{\star -1/\beta_1} \cdot \Upsilon_2^{\star -1/\beta_2}$ since, in this case, it was unfortunate when drawing the random index $j^\star$. Still, with probability $1/q_b$, the signature $\sigma^\star$ opens to the user introduced at the $j^\star$-th $Q_{\mathsf{b\text{-}join}}$-query and $(\Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star)$ does decrypt to $X^\star$. In this situation, the perfect soundness of the proof system ensures that $com^\star_{\sigma_{\mathsf{VK}^\star}}$ is a commitment to a group element $\sigma^\star_{\mathsf{VK}^\star}$ such that $e(\sigma^\star_{\mathsf{VK}^\star}, X^\dagger \cdot g^{\mathsf{VK}^\star}) = e(g,g)$. Since $\sigma^\star$ is a Type II forgery, $\mathcal{B}$ can use β_1, β_2 to compute a BBS decryption of $com^\star_{\sigma_{\mathsf{VK}^\star}}$ and obtain a pair of the form $(\sigma_{\mathsf{VK}^\star}, \mathsf{VK}^\star) = (g^{1/(x+\mathsf{VK}^\star)}, \mathsf{VK}^\star)$. The latter eventually yields a solution $(\tilde{g}^{1/(x+\mathsf{VK}^\star)}, \mathsf{VK}^\star)$ to the initial q_s-SDH instance by performing an Euclidean division in the exponent as in [12]. □

B.3 Anonymity

As for the anonymity property, it naturally relies on the DLIN assumption. The proof is essentially identical to that of Lemma 5 in [35] but we give it for completeness.

Theorem 4 (Anonymity). *The advantage of any anonymity adversary is at most*

$$\mathbf{Adv}^{\mathrm{anon}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathrm{ots}}(\lambda) + 3 \cdot \mathbf{Adv}^{\mathrm{DLIN}}(\lambda),$$

where the first term is $\mathcal{A}$'s probability of breaking the strong unforgeability of the one-time signature.

Proof. We consider a sequence of games at the end of which even an unbounded adversary has no advantage. In Game i, we call S_i the event that $\mathcal{A}$ wins and define $Adv_i = |\Pr[S_i] - 1/2|$.

Game 1: is the experiment of definition 8. In the play stage, the adversary $\mathcal{A}$ can obtain the group public key $\mathcal{Y}$, the group manager's private key $\mathcal{S}_{\mathsf{GM}} = (sk^{(0)}_{\mathsf{AHO}}, sk^{(1)}_{\mathsf{AHO}})$. It can also ask for the opening of any group signature and read/write the content of $\mathsf{state}_{\mathcal{I}}$. When it decides to enter the challenge phase, it outputs a message $M^\star$, a period index $t^\star$ and two membership certificate/secret $(\mathsf{cert}_0^\star, \mathsf{sec}_0^\star)$ and $(\mathsf{cert}_1^\star, \mathsf{sec}_1^\star)$ such that $\mathsf{cert}_b^\star \leftrightharpoons_{\mathcal{Y}} \mathsf{sec}_b^\star$ for $b = 0, 1$. The simulator $\mathcal{B}$ flips a fair coin $d \xleftarrow{R} \{0,1\}$ and computes $\sigma^\star \leftarrow \mathsf{Sign}(\mathcal{Y}, t^\star, RL_{t^\star}, \mathsf{cert}_d^\star, \mathsf{sec}_d^\star, M^\star)$, where $t^\star$ is determined by the history of Q_{revoke}-queries. The signature $\sigma^\star$ is given as a challenge to $\mathcal{A}$ who has to guess $d \in \{0,1\}$ after another series of queries (under the natural restriction of not querying the opening of $\sigma^\star$). We have $Adv_1 = \mathbf{Adv}^{\mathrm{anon}}(\mathcal{A})$.

Game 2: is as **Game** 1 but $\mathcal{B}$ halts if $\mathcal{A}$ queries the opening of a signature σ containing the same one-time verification key $\mathsf{VK}^\star$ as in the challenge phase (we assume w.l.o.g. that $(\mathsf{SK}^\star, \mathsf{VK}^\star)$ is generated at the outset of the game). If such a query is made before the challenge phase, it means that $\mathcal{A}$ was able to forge a one-time signature even without having seen a signature. If the query occurs after the challenge phase, then the strong unforgeability of Σ is broken. We can thus write

$|\Pr[S_2] - \Pr[S_1]| \leq \mathbf{Adv}^{\mathrm{ots}}(\lambda)$.

Game 3: we change the generation of $\mathcal{Y}$ so as to answer Q_{open}-queries without using the secret exponents $\beta_1, \beta_2 \in \mathbb{Z}_p$ that define $\mathcal{S}_{\mathsf{OA}}$. To this end, $\mathcal{B}$ chooses $\alpha_u, \alpha_v \xleftarrow{R} \mathbb{Z}_p^*$, and defines $U = g^{-\mathsf{VK}^\star} \cdot f_1^{\alpha_u}$, and $V = g^{-\mathsf{VK}^\star} \cdot f_2^{\alpha_v}$. It is not hard to see (see [42] for details) that, for any Q_{open}-query containing a BBS encryption $(\Upsilon_1, \Upsilon_2, \Upsilon_3) = (f_1^{z_1}, f_2^{z_2}, X \cdot g^{z_1+z_2})$, the values (Υ_4, Υ_5) reveal g^{z_1} and g^{z_2} (and thus the encrypted X) since $\mathsf{VK} \neq \mathsf{VK}^\star$ unless the event introduced in Game 2 occurs. To generate the challenge signature $\sigma^\star$ at epoch $t^\star$, the challenger $\mathcal{B}$ first computes $(\Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star)$ and then $(\Upsilon_4^\star, \Upsilon_5^\star) = (\Upsilon_1^{\star\alpha_u}, \Upsilon_2^{\star\alpha_v})$. It sets the challenge signature to be $\sigma^\star = (\mathsf{VK}^\star, \Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star, \Upsilon_4^\star, \Upsilon_5^\star, \Omega^\star, \mathbf{com}^\star, \mathbf{\Pi}^\star, \sigma_{ots}^\star)$. It can be checked that the distributions of $\mathcal{Y}$ and $\sigma^\star$ are unchanged and we have $\Pr[S_3] = \Pr[S_2]$.

Game 4: in the setup phase, we generate the CRS $\mathbf{f} = (\vec{f_1}, \vec{f_2}, \vec{f_3})$ of the proof system for the perfect WI setting. We choose $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2} \cdot (1, 1, g)^{-1}$ instead of $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$ so that $\vec{f_1}$, $\vec{f_2}$ and $\vec{f_3}$ are linearly independent. Any significant change in $\mathcal{A}$'s behavior yields a distinguisher for the DLIN problem and we can write $|\Pr[S_4] - \Pr[S_3]| = 2 \cdot \mathbf{Adv}^{\mathrm{DLIN}}(\mathcal{B})$. As noted in [36], proofs in the WI setting reveal no information on which witnesses they were generated from.

Game 5: in this game, we modify the generation of the challenge signature $\sigma^\star$ and use the trapdoor of the Groth-Sahai CRS (namely, the exponents ξ_1, ξ_2 for which $\vec{\varphi} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$) to generate simulated proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^3$ that $(\Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star)$ and com_X encrypt of the same value. It is known [36] that linear multi-exponentiation equations always have perfectly NIZK proofs on a simulated CRS. For, any satisfiable relation, (ξ_1, ξ_2) allows generating proofs without using the witnesses χ_1, χ_2, χ_3 for which (10) holds and simulated proofs are perfectly indistinguishable from real ones. Hence, $\Pr[S_5] = \Pr[S_4]$.

Game 6: in the computation of $\Upsilon_3^\star$, we now replace $g^{z_1+z_2}$ by a random group element in the challenge $\sigma^\star$. Since $\mathcal{B}$ does not explicitly use $z_1 = \log_{f_1}(\Upsilon_1^\star)$, $z_2 = \log_{f_2}(\Upsilon_2^\star)$, any change in $\mathcal{A}$'s behavior yields a distinguisher for the DLIN problem and $|\Pr[S_6] - \Pr[S_5]| \leq \mathbf{Adv}^{\mathrm{DLIN}}(\mathcal{B})$. In Game 6, we have $\Pr[S_6] = 1/2$. Indeed, when we consider the challenge $\sigma^\star$, Groth-Sahai commitments are all perfectly hiding in the WI setting and proofs $\mathbf{\Pi}$ reveal nothing about the underlying witnesses (in particular, NIZK proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^3$ are generated without using them) and $(\Upsilon_1^\star, \Upsilon_2^\star, \Upsilon_3^\star)$ perfectly hides $X^\star$. Finally, randomized signature components $\Omega^\star = \{\Theta'^{\star}_{l,i}, \theta'^{\star}_{l,i}\}_{i \in \{3,4,6,7\}}$ are information-theoretically independent of the corresponding messages and the remaining components of AHO signatures $\Theta_l^\star$ and $\theta_l^\star$.

When combining the above, $\mathcal{A}$'s advantage can be bounded by $\mathbf{Adv}^{\mathrm{anon}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathrm{ots}}(\lambda) + 3 \cdot \mathbf{Adv}^{\mathrm{DLIN}}(\lambda)$ as stated by the theorem. □

C Constructions from Weaker Assumptions

C.1 CDH-Based Vector Commitments

In [27], Catalano and Fiore described a vector commitment scheme whose binding property relies on the Diffie-Hellman assumption. In their scheme, if ℓ is the dimension of committed vectors, a commitment key

$$\big(g, g_1, \ldots, g_\ell, h_1, \ldots, h_\ell, \{h_{i,j}\}_{i \neq j}^{\ell}\big) \in \mathbb{G}^{1+\ell+\ell^2}$$

is obtained by randomly choosing $\alpha_1, \ldots, \alpha_\ell \xleftarrow{R} \mathbb{Z}_p^*$ and defining $g_i = g^{\alpha_i}$, $h_i = g^{\prod_{\kappa \neq i} \alpha_\kappa}$ and $h_{i,j} = g^{\prod_{\kappa \neq i,j}^{\ell} \alpha_\kappa} = h_j^{1/\alpha_i}$ (so that $h_{i,j} = h_{i,j}$) for each $i \in \{1, \ldots, \ell\}$ and $j \neq i$. A commitment to

$\vec{m} = (m_1, \ldots, m_\ell)$ is obtained as $C = \prod_{\kappa=1}^{\ell} g_\kappa^{m_\kappa}$. By revealing $W_i = \prod_{\kappa=1,\ \kappa\neq i}^{\ell} h_{i,\kappa}^{m_\kappa}$, the committer can open the commitment to m_i at the i-th coordinate of $\vec{m}$ as it satisfies the equation

$$e(g, C) \cdot e(g^{-m_i}, h_i) = e(g_i, W_i).$$

This time, the coordinate-wise binding property relies on the standard Computational Diffie-Hellman (CDH) assumption. Note that, in its basic version, the commitment is not (and does not need to be) hiding since it does not use any randomizer.

C.2 Construction

This section gives an alternative construction of revocable group signature where the ℓ-FlexDHE assumption is not used. Instead, we rely on an assumption (suggested in [44]) of fixed size, which is inspired by the Flexible Diffie-Hellman assumption [43].

Definition 9 ([44]). *In a group $\mathbb{G}$ of prime order p, the Flexible Square Diffie-Hellman (FSDH) problem consists in, given (g, g^a) with $a \xleftarrow{R} \mathbb{Z}_p$, finding a non-trivial triple $(g^\mu, g^{a\cdot\mu}, g^{(a^2)\cdot\mu})$, with $\mu \neq 0$.*

The Flexible Square Diffie-Hellman assumption is the hardness of FSDH for any PPT algorithm.

We thus trade one of the q-type assumptions for a constant-size assumption at the cost of increasing the size of the group public key. Indeed, the latter now contains $O(\log^2 N)$ group elements.

Setup(λ, N)**:** given a security parameter $\lambda \in \mathbb{N}$ and the maximal number of users $N = 2^{\ell-1}$,

1. Choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$, with a generator $g \xleftarrow{R} \mathbb{G}$.
2. Define $n_0 = 2$ and $n_1 = 7$. Generate two key pairs $(sk_{\mathsf{AHO}}^{(0)}, pk_{\mathsf{AHO}}^{(0)})$ and $(sk_{\mathsf{AHO}}^{(1)}, pk_{\mathsf{AHO}}^{(1)})$ for the AHO signature in order to sign messages of n_0 and n_1 group elements, respectively. These key pairs are

$$pk_{\mathsf{AHO}}^{(d)} = \Big(G_r^{(d)},\ H_r^{(d)},\ G_z^{(d)} = G_r^{\gamma_z^{(d)}},\ H_z^{(d)} = H_r^{\delta_z^{(d)}},\ \{G_i^{(d)} = G_r^{\gamma_i^{(d)}}, H_i^{(d)} = H_r^{\delta_i^{(d)}}\}_{i=1}^{n_d},\ A^{(d)},\ B^{(d)}\Big)$$

   and $sk_{\mathsf{AHO}}^{(d)} = \big(\alpha_a^{(d)}, \alpha_b^{(d)}, \gamma_z^{(d)}, \delta_z^{(d)}, \{\gamma_i^{(d)}, \delta_i^{(d)}\}_{i=1}^{n_d}\big)$, where $d \in \{0, 1\}$. These two schemes will be used to sign messages consisting of 2 and 7 group elements, respectively.
3. Generate a public key $ck = \big(g_1, \ldots, g_\ell, h_1, \ldots, h_\ell, \{h_{i,j}\}_{i\neq j}^{\ell}\big) \in \mathbb{G}^{\ell+\ell^2}$ for vectors of dimension ℓ in the CDH-based vector commitment scheme recalled in Section C.2.
4. As a CRS for the NIWI proof system, select vectors $\mathbf{f} = (\vec{f_1}, \vec{f_2}, \vec{f_3})$ s.t. $\vec{f_1} = (f_1, 1, g) \in \mathbb{G}^3$, $\vec{f_2} = (1, f_2, g) \in \mathbb{G}^3$, and $\vec{f_3} = \vec{f_1}^{\xi_1} \cdot \vec{f_2}^{\xi_2}$, with $f_1 = g^{\beta_1}$, $f_2 = g^{\beta_2} \xleftarrow{R} \mathbb{G}$ and $\beta_1, \beta_2, \xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$. We also define the vector $\vec{\varphi} = \vec{f_3} \cdot (1, 1, g)$.
5. Choose $(U, V) \xleftarrow{R} \mathbb{G}^2$ that, together with generators $f_1, f_2, g \in \mathbb{G}$, will form a public encryption key.
6. Select a strongly unforgeable one-time signature $\Sigma = (\mathcal{G}, \mathcal{S}, \mathcal{V})$.
7. Set $\mathcal{S}_{\mathsf{GM}} := \big(sk_{\mathsf{AHO}}^{(0)}, sk_{\mathsf{AHO}}^{(1)}\big)$, $\mathcal{S}_{\mathsf{OA}} := \big(\beta_1, \beta_2\big)$ as authorities' private keys and the group public key is

$$\mathcal{Y} := \Big(g,\ pk_{\mathsf{AHO}}^{(0)},\ pk_{\mathsf{AHO}}^{(1)},\ ck = \big(g_1, \ldots, g_\ell, h_1, \ldots, h_\ell, \{h_{i,j}\}_{i\neq j}^{\ell}\big),\ \mathbf{f},\ \vec{\varphi},\ (U, V),\ \Sigma\Big).$$

Join$^{(\mathrm{GM},\mathcal{U}_i)}$**:** the group manager and the prospective user $\mathcal{U}_i$ run the following interactive protocol $[\mathsf{J}_{\mathsf{user}}(\lambda,\mathcal{Y}),\mathsf{J}_{\mathsf{GM}}(\lambda,St,\mathcal{Y},\mathcal{S}_{\mathsf{GM}})]$:

1. $\mathsf{J}_{\mathsf{user}}(\lambda,\mathcal{Y})$ picks $x \xleftarrow{R} \mathbb{Z}_p$ and computes $X = g^x$ which is sent to $\mathsf{J}_{\mathsf{GM}}(\lambda,St,\mathcal{Y},\mathcal{S}_{\mathsf{GM}})$. If $X \in \mathbb{G}$ already appears in the database St_{trans}, J_{GM} halts and returns $\perp$ to $\mathsf{J}_{\mathsf{user}}$.
2. J_{GM} assigns to $\mathcal{U}_i$ an available leaf v of identifier $\mathsf{ID}(v)$ in the tree T. Let $x_1,\ldots,x_\ell$ be the path from $x_\ell = v$ to the root $x_1 = \epsilon$ of T. Let also $(I_1,\ldots,I_\ell) = (\mathsf{ID}(x_1),\ldots,\mathsf{ID}(x_\ell))$ be the vector of identifiers (with $I_1 = 1$ and $I_\ell = \mathsf{ID}(v) \in \{N,\ldots,2N-1\}$). Then, J_{GM} conducts the following steps.
   a. Compute a compact encoding of $(I_1,\ldots,I_\ell)$ as $C_v = \prod_{\kappa=1}^{\ell} g_\kappa^{I_\kappa} \in \mathbb{G}$.
   b. Using $sk_{\mathsf{AHO}}^{(0)}$, generate an AHO signature $\sigma_v = (\theta_{v,1},\ldots,\theta_{v,7})$ on $(X,C_v) \in \mathbb{G}^2$ in order to bind C_v to the value X that identifies the new member $\mathcal{U}_i$.
3. J_{GM} sends $\mathsf{ID}(v) \in \{N,\ldots,2N-1\}$ and C_v to $\mathsf{J}_{\mathsf{user}}$ that halts if $\mathsf{ID}(v) \notin \{N,\ldots,2N-1\}$ or if $C_v \neq \prod_{\kappa=1}^{\ell} g_\kappa^{I_\kappa} \in \mathbb{G}$. Otherwise, $\mathsf{J}_{\mathsf{user}}$ sends a signature $sig_i = \mathrm{Sign}_{\mathsf{usk}[i]}\big(X||(I_1,\ldots,I_\ell)\big)$ to J_{GM}.
4. J_{GM} checks that $\mathrm{Verify}_{\mathsf{upk}[i]}\big((X||(I_1,\ldots,I_\ell)),sig_i\big) = 1$. If not J_{GM} aborts. Otherwise, J_{GM} returns σ_v to $\mathsf{J}_{\mathsf{user}}$ and stores $\mathsf{transcript}_i = (X,\mathsf{ID}(v),C_v,\sigma_v,sig_i)$ in the database St_{trans}.
5. $\mathsf{J}_{\mathsf{user}}$ defines the membership certificate as $\mathsf{cert}_i = \big(\mathsf{ID}(v),X,C_v,\sigma_v\big) \in \{N,\ldots,2N-1\}\times\mathbb{G}^9$, where X will serve as the tag identifying $\mathcal{U}_i$. The membership secret sec_i is defined as $\mathsf{sec}_i = x \in \mathbb{Z}_p$.

Revoke$(\mathcal{Y},\mathcal{S}_{\mathsf{GM}},t,\mathcal{R}_t)$**:** Parse $\mathcal{S}_{\mathsf{GM}}$ as $\mathcal{S}_{\mathsf{GM}} := \big(sk_{\mathsf{AHO}}^{(0)},sk_{\mathsf{AHO}}^{(1)}\big)$ and do the following.

1. Find a partition of the unrevoked user set $\{1,\ldots,N\}\backslash\mathcal{R}_t$ as the union of disjoint subsets of the form $S_{k_1,u_1},\ldots,S_{k_m,u_m}$, with $m \leq 2\cdot|\mathcal{R}_t|-1$.
2. For $i = 1$ to m, do the following.
   a. Parse S_{k_i,u_i} as the difference between sub-trees rooted at an internal node x_{k_i} and one of its descendants x_{u_i}. Let $\phi_i,\psi_i \in \{1,\ldots,\ell\}$ be the depths of x_{k_i} and x_{u_i}, respectively, in T assuming that the root ϵ is at depth 1. Encode S_{k_i,u_i} as a vector of group elements
   $$\big(g_{\phi_i},h_{\phi_i},g^{-\mathsf{ID}(x_{k_i})},g_{\psi_i},h_{\psi_i},g^{\mathsf{ID}(x_{u_i})}\big) \in \mathbb{G}^6.$$
   b. To authenticate S_{k_i,u_i} and link it to the revocation epoch t, use $sk_{\mathsf{AHO}}^{(1)}$ to compute a structure-preserving signature $\Theta_i = (\Theta_{i,1},\ldots,\Theta_{i,7}) \in \mathbb{G}^7$ on the message
   $$R_i = \big(g^t,g_{\phi_i},h_{\phi_i},g^{-\mathsf{ID}(x_{k_i})},g_{\psi_i},h_{\psi_i},g^{\mathsf{ID}(x_{u_i})}\big) \in \mathbb{G}^7,$$
   where the epoch number t is interpreted as an element of $\mathbb{Z}_p$.

Return
$$RL_t = \Big(t,\ \mathcal{R}_t,\ \{\phi_i,\ \psi_i,\ \mathsf{ID}(x_{k_i}),\ \mathsf{ID}(x_{u_i}),\ \Theta_i = (\Theta_{i,1},\ldots,\Theta_{i,7})\}_{i=1}^{m}\Big). \tag{24}$$

Sign$(\mathcal{Y},t,RL_t,\mathsf{cert}_i,\mathsf{sec}_i,M)$**:** return $\perp$ if $i \in \mathcal{R}_t$. Otherwise, to sign M, generate a one-time key pair $(\mathsf{SK},\mathsf{VK}) \leftarrow \mathcal{G}(\lambda)$. Parse cert_i as $\mathsf{cert}_i = \big(\mathsf{ID}(v_i),X,C_{v_i},\sigma_{v_i}\big) \in \{N,\ldots,2N-1\}\times\mathbb{G}^9$ and sec_i as $x \in \mathbb{Z}_p$. Let $\epsilon = x_1,\ldots,x_\ell = v_i$ be the path connecting the leaf v_i to the root ϵ and let $(I_1,\ldots,I_\ell) = (\mathsf{ID}(x_1),\ldots,\mathsf{ID}(x_\ell))$. First, $\mathcal{U}_i$ generates a commitment $com_{C_{v_i}}$ to the encoding C_{v_i} of the path $(I_1,\ldots,I_\ell)$ from v_i to the root. Then, conduct the following steps.

1. Using RL_t, find the set S_{k_l,u_l}, with $l \in \{1,\ldots,m\}$, that contains the leaf v_i identified by v_i. Let x_{k_l} and x_{u_l} denote the primary and secondary roots of S_{k_l,u_l} at depths ϕ_l and ψ_l, respectively. Since x_{k_l} is an ancestor of v_i but x_{u_l} is not, it holds that $I_{\phi_l} = \mathsf{ID}(x_{k_l})$ and $I_{\psi_l} \neq \mathsf{ID}(x_{u_l})$.

2. To prove that v_i belongs to S_{k_l,u_l}, $\mathcal{U}_i$ first re-randomizes the l-th AHO signature Θ_l of RL_t as $\{\Theta'_{l,i}\}_{i=1}^{7} \leftarrow \mathsf{ReRand}(pk_{\mathsf{AHO}}^{(1)}, \Theta_l)$. Then, he commits to the l-th revocation message

$$R_l = (R_{l,1}, R_{l,2}, R_{l,3}, R_{l,4}, R_{l,5}, R_{l,6}, R_{l,7}) = \big(g^t,\ g_{\phi_l},\ h_{\phi_l},\ g^{-\mathsf{ID}(x_{k_l})},\ g_{\psi_l},\ h_{\psi_l},\ g^{\mathsf{ID}(x_{u_l})}\big) \quad (25)$$

and its signature $\Theta'_l = (\Theta'_{l,1}, \ldots, \Theta'_{l,7})$ by computing Groth-Sahai commitments $\{com_{R_{l,\tau}}\}_{\tau=2}^{7}$, $\{com_{\Theta'_{l,j}}\}_{j\in\{1,2,5\}}$ to $\{R_{l,\tau}\}_{\tau=2}^{7}$ and $\{\Theta'_{l,j}\}_{j\in\{1,2,5\}}$.

   a. To prove that $I_{\phi_l} = \mathsf{ID}(x_{k_l})$, $\mathcal{U}_i$ first computes $W_{\phi_l} = \prod_{\kappa=1,\ \kappa\neq\phi_l}^{\ell} h_{\phi_l,\kappa}^{I_\kappa}$ that satisfies the equality $e(g, C_{v_i}) \cdot e(g^{-I_{\phi_l}}, h_{\phi_l}) = e(g_{\phi_l}, W_{\phi_l})$. Then, $\mathcal{U}_i$ generates a Groth-Sahai commitment $com_{W_{\phi_l}}$ to W_{ϕ_l}. He computes a proof that committed variables $(R_{l,2}, R_{l,3}, R_{l,4}, C_{v_i}, W_{\phi_l})$ satisfy the equation

$$e(g, C_{v_i}) \cdot e(R_{l,4}, R_{l,3}) = e(R_{l,2}, W_{\phi_l}). \quad (26)$$

   Let π_{eq} be the proof for the quadratic equation (26).

   b. To prove that $I_{\psi_l} \neq \mathsf{ID}(x_{u_l})$, $\mathcal{U}_i$ computes $W_{\psi_l} = \prod_{\kappa=1,\ \kappa\neq\psi_l}^{\ell} h_{\psi_l,\kappa}^{I_\kappa}$ that satisfies the equality $e(g, C_{v_i}) \cdot e(g^{-I_{\psi_l}}, h_{\psi_l}) = e(g_{\psi_l}, W_{\psi_l})$. Then, he computes a commitment $com_{W_{\psi_l}}$ to W_{ψ_l} as well as commitments com_{Γ_l} and $\{com_{\Psi_{l,\tau}}\}_{\tau=0,1}$ to the group elements

$$(\Gamma_l, \Psi_{l,0}, \Psi_{l,1}) = \big(g^{1/(\mathsf{ID}(x_{u_l})-I_{\psi_l})}, g^{-I_{\psi_l}}, g_{\psi_l}^{-I_{\psi_l}}\big).$$

   Then, $\mathcal{U}_i$ provides evidence that committed variables $(R_{l,5}, R_{l,6}, R_{l,7}, C_{v_i}, \Gamma_l, \Psi_l)$ satisfy

$$e(g, C_{v_i}) \cdot e(\Psi_{l,0}, R_{l,6}) = e(R_{l,5}, W_{\phi_l}), \quad (27)$$
$$e(R_{l,7} \cdot \Psi_{l,0}, \Gamma_l) = e(g, g) \quad (28)$$
$$e(\Psi_{l,0}, R_{l,5}) = e(g, \Psi_{l,1}). \quad (29)$$

   We denote this proof by $\pi_{neq} = (\pi_{neq,1}, \pi_{neq,2}, \pi_{neq,3})$. It consists of 27 group elements since all equations are quadratic.

3. $\mathcal{U}_i$ proves that the tuple R_l of (25) is part of RL_t: namely, $\mathcal{U}_i$ computes a proof π_{R_l} that committed message elements $\{R_{l,\tau}\}_{\tau=2}^{7}$ and signature components $\{\Theta'_{l,j}\}_{j\in\{1,2,5\}}$ satisfy the equations

$$A^{(1)} \cdot e(\Theta'_{l,3}, \Theta'_{l,4})^{-1} \cdot e(G_1^{(1)}, g^t)^{-1} = e(G_z^{(1)}, \Theta'_{l,1}) \cdot e(G_r^{(1)}, \Theta'_{l,2}) \cdot \prod_{\tau=2}^{7} e(G_\tau^{(1)}, R_{l,\tau}), \quad (30)$$
$$B^{(1)} \cdot e(\Theta'_{l,6}, \Theta'_{l,7})^{-1} \cdot e(H_1^{(1)}, g^t)^{-1} = e(H_z^{(1)}, \Theta'_{l,1}) \cdot e(H_r^{(1)}, \Theta'_{l,5}) \cdot \prod_{\tau=2}^{7} e(H_\tau^{(1)}, R_{l,\tau}),$$

   The proof π_{R_l} takes 6 elements as both equations of (30) are linear.

4. Let $\sigma_{v_i} = (\theta_{v_i,1}, \ldots, \theta_{v_i,7})$ be the AHO signature on the message (X, C_{v_i}). Set $\{\theta'_{v_i,j}\}_{j=1}^{7} \leftarrow \mathsf{ReRand}(pk_{\mathsf{AHO}}^{(0)}, \sigma_{v_i})$ and generate commitments $\{com_{\theta'_{v_i,j}}\}_{j\in\{1,2,5\}}$ to $\{\theta'_{v_i,j}\}_{j\in\{1,2,5\}}$ as well as a commitment com_X to X. Then, generate a proof $\pi_{\sigma_{v_i}}$ that committed variables satisfy

$$A^{(0)} \cdot e(\theta'_{l,3}, \theta'_{l,4})^{-1} = e(G_z^{(0)}, \theta'_{l,1}) \cdot e(G_r^{(0)}, \theta'_{l,2}) \cdot e(G_1^{(0)}, X) \cdot e(G_2^{(0)}, C_{v_i}),$$
$$B^{(0)} \cdot e(\theta'_{l,6}, \theta'_{l,7})^{-1} = e(H_z^{(0)}, \theta'_{l,1}) \cdot e(H_r^{(0)}, \theta'_{l,5}) \cdot e(H_1^{(0)}, X) \cdot e(H_2^{(0)}, C_{v_i})$$

Since these equations are linear, $\pi_{\sigma_{v_i}}$ requires 6 group elements.

5. Using VK as a tag, compute a tag-based encryption [42] of X by picking $z_1, z_2 \xleftarrow{R} \mathbb{Z}_p$ and setting
$$(\Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5) = \big(f_1^{z_1}, f_2^{z_2}, X \cdot g^{z_1+z_2}, (g^{\mathsf{VK}} \cdot U)^{z_1}, (g^{\mathsf{VK}} \cdot V)^{z_2}\big).$$
6. Generate a NIZK proof that $com_X = (1, 1, X) \cdot \vec{f}_1^{\,w_{X,1}} \cdot \vec{f}_2^{\,w_{X,2}} \cdot \vec{f}_3^{\,w_{X,3}}$ and $(\Upsilon_1, \Upsilon_2, \Upsilon_3)$ are BBS encryptions of the same value X. If we write $\vec{f}_3 = (f_{3,1}, f_{3,2}, f_{3,3})$, the Groth-Sahai commitment com_X can be written as $(f_1^{w_{X,1}} \cdot f_{3,1}^{w_{X,3}}, f_2^{w_{X,2}} \cdot f_{3,2}^{w_{X,3}}, X \cdot g^{w_{X,1}+w_{X,2}} \cdot f_{3,3}^{w_{X,3}})$, so that we have
$$com_X \cdot (\Upsilon_1, \Upsilon_2, \Upsilon_3)^{-1} = \big(f_1^{\chi_1} \cdot f_{3,1}^{\chi_3},\ f_2^{\chi_2} \cdot f_{3,2}^{\chi_3},\ g^{\chi_1+\chi_2} \cdot f_{3,3}^{\chi_3}\big) \tag{31}$$
with $\chi_1 = w_{X,1} - z_1$, $\chi_2 = w_{X,2} - z_2$, $\chi_3 = w_{X,3}$. The signer $\mathcal{U}_i$ commits to $\chi_1, \chi_2, \chi_3 \in \mathbb{Z}_p$ (by computing com_{χ_j}, for $j \in \{1, 2, 3\}$), and generates proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^{3}$ that χ_1, χ_2, χ_3 satisfy the relations (31).
7. Compute a weak Boneh-Boyen signature $\sigma_{\mathsf{VK}} = g^{1/(x+\mathsf{VK})}$ on VK and a commitment $com_{\sigma_{\mathsf{VK}}}$ to σ_{VK}. Then, generate a NIWI proof $\pi_{\sigma_{\mathsf{VK}}} = (\vec{\pi}_{\sigma_{\mathsf{VK}},1}, \vec{\pi}_{\sigma_{\mathsf{VK}},2}, \vec{\pi}_{\sigma_{\mathsf{VK}},3}) \in \mathbb{G}^9$ that committed variables $(\sigma_{\mathsf{VK}}, X) \in \mathbb{G}^2$ satisfy the quadratic equation $e(\sigma_{\mathsf{VK}}, X \cdot g^{\mathsf{VK}}) = e(g, g)$.
8. Compute $\sigma_{ots} = \mathcal{S}(\mathsf{SK}, (M, RL_t, \Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5, \Omega, \mathbf{com}, \mathbf{\Pi}))$ where $\Omega = \{\Theta'_{l,i}, \theta'_{l,i}\}_{i\in\{3,4,6,7\}}$ and

$$\mathbf{com} = \big(com_{C_{v_i}}, com_X, \{com_{R_{l,\tau}}\}_{\tau=2}^{7}, com_{W_{\phi_l}}, com_{W_{\psi_l}}, com_{\Gamma_l},$$
$$\{com_{\Psi_{l,\tau}}\}_{\tau\in\{0,1\}}, \{com_{\Theta'_{l,j}}\}_{j\in\{1,2,5\}}, \{com_{\theta'_{l,j}}\}_{j\in\{1,2,5\}}, \{com_{\chi_j}\}_{j=1}^{3}, com_{\sigma_{\mathsf{VK}}}\big)$$
$$\mathbf{\Pi} = \big(\pi_{eq}, \pi_{neq}, \pi_{R_l}, \pi_{\sigma_{v_i}}, \{\pi_{eq\text{-}com,j}\}_{j=1}^{3}, \pi_{\sigma_{\mathsf{VK}}}\big)$$

Return the signature $\sigma = \big(\mathsf{VK}, \Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5, \Omega, \mathbf{com}, \mathbf{\Pi}, \sigma_{ots}\big)$.

Verify$(\sigma, M, t, RL_t, \mathcal{Y})$**:** parse σ as above and do the following.

1. If $\mathcal{V}(\mathsf{VK}, (M, RL_t, \Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4, \Upsilon_5, \Omega, \mathbf{com}, \mathbf{\Pi}), \sigma_{ots}) = 0$, return 0.
2. Return 0 if $e(\Upsilon_1, g^{\mathsf{VK}} \cdot U) \neq e(f_1, \Upsilon_4)$ or $e(\Upsilon_2, g^{\mathsf{VK}} \cdot V) \neq e(f_2, \Upsilon_5)$.
3. Return 1 if all proofs properly verify. Otherwise, return 0.

Open$(M, t, RL_t, \sigma, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St)$**:** parse σ as above and return $\perp$ if $\mathsf{Verify}(\sigma, M, t, RL_t, \mathcal{Y}) = 0$. Otherwise, given $\mathcal{S}_{\mathsf{OA}} = (\beta_1, \beta_2)$, compute $\tilde{X} = \Upsilon_3 \cdot \Upsilon_1^{-1/\beta_1} \cdot \Upsilon_2^{-1/\beta_2}$. In the database St_{trans}, find a record $\langle i, \mathsf{transcript}_i = (X_i, \mathsf{ID}(v_i), C_{v_i}, \sigma_{v_i}, sig_i)\rangle$ such that $X_i = \tilde{X}$. If no such record exists in St_{trans}, return $\perp$. Otherwise, return i.

Each signature now consists of 150 group elements since **com** and $\mathbf{\Pi}$ contain 69 and 63 group elements, respectively. The only overhead is in the size of the group public key which grows from $O(\log N)$ to $O(\log^2 N)$.

C.3 Security

Theorem 5 (Misidentification). *The scheme is secure against misidentification attacks assuming that the q-SFP and the FSDH problems are both hard for $q = \max(q_a, q_r^2)$, where q_a and q_r denote the maximal numbers of $Q_{\mathsf{a\text{-}join}}$ queries and Q_{revoke} queries, respectively, and N is the maximal number of group members.*

Proof. The proof is almost identical to the proof of Theorem 2. It considers the same two kinds of forgeries and the only difference is the treatment of Type II.b forgeries. Lemma 3 shows how to break the 2-3-SqDH assumption using a Type II.b forger. □

Lemma 3. *The advantage of any Type II.b forger $\mathcal{A}$ is at most $\mathbf{Adv}_{\mathcal{A}}^{\text{mis-id-II.b}}(\lambda) \leq \ell \cdot \mathbf{Adv}^{\text{FSDH}}(\lambda)$, where $\ell = \log N$ and N is the maximal number of users.*

Proof. To prove the result, it is convenient to use an equivalent formulation[3] of the problem. Namely, given (g, g^a), we have to find a triple $(g^{a\cdot\mu}, g^{\mu}, g^{\mu/a})$ for some $\mu \neq 0$. We describe an algorithm $\mathcal{B}$ that receives as input an instance $(g, g^a) \in \mathbb{G}^2$ of the FSDH problem and uses the Type II.b forger to find a non-trivial $(g^{a\cdot\mu}, g^{\mu}, g^{\mu/a})$. To generate the group public key, $\mathcal{B}$ follows the specification of the Setup procedure except that, instead of computing ck as in step 3 of the algorithm, it defines $ck = (g_1, \ldots, g_\ell, h_1, \ldots, h_\ell, \{h_{i,j}\}_{i\neq j})$ as follows. It picks $i^\star \xleftarrow{R} \{1, \ldots, \ell\}$ and defines

$$
\begin{aligned}
g_{i^\star} &= g^a && \\
g_i &= g^{z_i} && i \neq i^\star \\
h_{i^\star} &= g^{\prod_{\kappa\neq i^\star} z_\kappa} && \\
h_i &= (g^a)^{\prod_{\kappa\neq i,i^\star} z_\kappa} && i \neq i^\star \\
h_{ij} &= (g^a)^{\prod_{\kappa\neq i,j,i^\star} z_\kappa} && i \neq i^\star,\ j \neq i^\star \\
h_{i^\star j} &= g^{\prod_{\kappa\neq j,i^\star} z_\kappa} && j \neq i^\star \\
h_{ii^\star} &= g^{\prod_{\kappa\neq i,i^\star} z_\kappa} && i \neq i^\star
\end{aligned}
$$

where $z_1, \ldots, z_\ell \xleftarrow{R} \mathbb{Z}_p$. Eventually, $\mathcal{Y} := \big(g, pk_{\mathsf{AHO}}^{(0)}, pk_{\mathsf{AHO}}^{(1)}, ck, \mathbf{f}, \vec{\varphi}, (U, V), \Sigma\big)$ is given to the Type II.b forger $\mathcal{A}$.

During the whole game, the adversary can adaptively probe the Q_{pub}, $Q_{\mathsf{a\text{-}join}}$, Q_{revoke}, Q_{read}, and Q_{keyOA} oracles. Since $\mathcal{S}_{\mathsf{GM}} = (sk_{\mathsf{AHO}}^{(0)}, sk_{\mathsf{AHO}}^{(1)})$ and $\mathcal{S}_{\mathsf{OA}} = (\beta_1, \beta_2)$ are available to the reduction $\mathcal{B}$, the latter can always perfectly answer adversarial queries. At the end of the game, the adversary $\mathcal{A}$ outputs a forgery $\sigma^\star$ for which the committed variables $C_{v_i}^\star$, $(R_{l,2}^\star, \ldots, R_{l,7}^\star)$, $(\Psi_{l,0}^\star, \Psi_{l,1}^\star)$ and $(\Gamma_l^\star, W_{\phi_l}^\star, W_{\psi_l}^\star)$ satisfy the relations

$$e(g, C_{v_i}^\star) \cdot e(R_{l,4}^\star, R_{l,3}^\star) = e(R_{l,2}^\star, W_{\phi_l}^\star) \tag{32}$$

$$e(g, C_{v_i}^\star) \cdot e(\Psi_{l,0}^\star, R_{l,6}^\star) = e(R_{l,5}^\star, W_{\psi_l}^\star), \tag{33}$$

$$e(R_{l,7}^\star \cdot \Psi_{l,0}^\star, \Gamma_l^\star) = e(g, g), \tag{34}$$

$$e(\Psi_{l,0}^\star, R_{l,5}^\star) = e(g, \Psi_{l,1}^\star). \tag{35}$$

[3] Given (g, g^a), if we define $y = g^a$ and $y^A = g$ (so that $A = 1/a$) any FSDH solution $(y^\mu, y^{A\cdot\mu}, y^{(A^2)^\mu})$ can be written as $(g^{a\cdot\mu}, g^{\mu}, g^{\mu/a})$

although $\sigma^\star$ opens to some user $i^\star \in U^a \cap \mathcal{R}_{t^\star}$.

Note that $(R^\star_{l,1}, \ldots, R^\star_{l,7})$ is necessarily of the form

$$(R^\star_{l,1}, R^\star_{l,2}, R^\star_{l,3}, R^\star_{l,4}, R^\star_{l,5}, R^\star_{l,6}, R^\star_{l,7}) = \Big(g^{t^\star},\ g_{\phi_l},\ h_{\phi_l},\ g^{-\mathsf{ID}(x^\star_{k_l})},\ g_{\psi_l},\ h_{\psi_l},\ g^{\mathsf{ID}(x^\star_{k_l})}\Big), \tag{36}$$

for some indices $\phi_l, \psi_l \in \{1, \ldots, \ell\}$ and some node identifiers $\mathsf{ID}(x^\star_{k_l}), \mathsf{ID}(x^\star_{u_l}) \in \{1, \ldots, 2N-1\}$ that were chosen by $\mathcal{B}$ at the latest Q_{revoke}-query. Since, by hypothesis, $\sigma^\star$ contains a committed pair $(X^\star, C^\star_{v_i})$ that was signed by $\mathcal{B}$ during some $Q_{\mathsf{a\text{-}join}}$-query, $\mathcal{B}$ also knows $(I^\star_1, \ldots, I^\star_\ell)$ such that $C^\star_{v_i} = \prod_{\kappa=1}^{\ell} g_\kappa^{I^\star_\kappa}$. Since $i^\star \in U^a \cap \mathcal{R}_{t^\star}$, it must hold that either:

- $I^\star_{\phi_l} \neq \mathsf{ID}(x^\star_{k_l})$: In this case, relations (36) and (32) imply that

$$e(g, C^\star_{v_i}) \cdot e(g^{-\mathsf{ID}(x^\star_{k_l})}, h_{\phi_\ell}) = e(g_{\phi_l}, W^\star_{\phi_l}) \tag{37}$$

  for values $\phi_l \in \{1, \ldots, \ell\}$ and $\mathsf{ID}(x_{k_l})^\star \in \{1, \ldots, 2N-1\}$ that are available to $\mathcal{B}$. At this point, $\mathcal{B}$ fails if $\phi_l \neq i^\star$. With probability $1/\ell$ however, it holds that $\phi_l = i^\star$ in which case $\mathcal{B}$ can solve the problem as follows. Since it knows $(I^\star_1, \ldots, I^\star_\ell)$ such that $C^\star_{v_i} = \prod_{\kappa=1}^{\ell} g_\kappa^{I^\star_\kappa}$, it can compute $W' = \prod_{\kappa=1,\ \kappa\neq\phi_l}^{\ell} h_{\phi_l,\kappa}^{I^\star_\kappa}$ which satisfies

$$e(g, C^\star_{v_i}) \cdot e(g^{-I^\star_{\phi_l}}, h_{\phi_\ell}) = e(g_{\phi_l}, W'_{\phi_l}) \tag{38}$$

  By dividing (37) and (38), we find that $e(g_{i^\star}, (W^\star_{\phi_l}/W'_{\phi_l})^{1/(I^\star_{\phi_l} - \mathsf{ID}(x^\star_{k_l}))}) = e(g, h_{i^\star})$. This implies that, by computing $g^{1/a} = \big(W^\star_{\phi_l}/W'\big)^{1/(I^\star_{\phi_l} - \mathsf{ID}(x^\star_{k_l}))\cdot\prod_{\kappa\neq i^\star} z_\kappa}$, $\mathcal{B}$ actually solves a problem which is at least as hard as FSDH.

- $I^\star_{\psi_l} = \mathsf{ID}(x^\star_{u_l})$: If we define $\varrho = -\log_g(\Psi^\star_{l,0})$, relations (36) and (33)-(35) imply that

$$e(g, C^\star_{v_i}) \cdot e(g^{-\varrho}, h_{\psi_\ell}) = e(g_{\psi_l}, W'_{\psi_l}) \tag{39}$$

$$g^{\varrho - I^\star_{\psi_l}} \neq 1_{\mathbb{G}} \tag{40}$$

$$\Psi^\star_{l,0} = g^{-\varrho} \tag{41}$$

$$\Psi^\star_{l,1} = g^{-\varrho}_{\psi_l}, \tag{42}$$

  for some $\psi_l \in \{1, \ldots, \ell\}$. At this point, $\mathcal{B}$ halts and declares failure if $\psi_l \neq i^\star$. Still, with probability $1/\ell$, we have $\psi_l = i^\star$ and $\mathcal{B}$ can solve the 2-3-SqDH as follows. Similarly to the previous case, it can compute $W' = \prod_{\kappa=1,\ \kappa\neq\psi_l}^{\ell} h_{\psi_l,\kappa}^{I^\star_\kappa}$ such that

$$e(g, C^\star_{v_i}) \cdot e(g^{-I^\star_{\psi_l}}, h_{\psi_\ell}) = e(g_{\psi_l}, W^\star_{\psi_l}) \tag{43}$$

  Now, by dividing (39) from (43), we obtain the equality $e(g, h_{\psi_l})^{\varrho - I^\star_{\psi_l}} = e(g_{\psi_l}, W'/W^\star_{\psi_l})$ which, if $\psi = i^\star$, implies $W'/W^\star_{\phi_l} = g^{(\varrho - I^\star_{\psi_l})\cdot\prod_{\kappa\neq i^\star} z_\kappa/a}$. The triple

$$\Big((\Psi^{\star\,-1}_{l,1} \cdot (g^a)^{-I^\star_{\psi_l}})^{\prod_{\kappa\neq i^\star} z_\kappa},\ (\Psi^{\star\,-1}_{l,0} \cdot g^{-I^\star_{\psi_l}})^{\prod_{\kappa\neq i^\star} z_\kappa},\ W'/W^\star_{\phi_l}\Big)$$
$$= \Big(g^{a\left(\varrho - I^\star_{\psi_l}\right)\cdot\prod_{\kappa\neq i^\star} z_\kappa},\ g^{\left(\varrho - I^\star_{\psi_l}\right)\cdot\prod_{\kappa\neq i^\star} z_\kappa},\ g^{\frac{(\varrho - I^\star_{\psi_l})\cdot\prod_{\kappa\neq i^\star} z_\kappa}{a}}\Big)$$

  is a non-trivial solution to the FSDH instance.

In both cases, we observe that, if $\mathcal{A}$ is able to mount a Type II.b attack with probability ε, then $\mathcal{B}$ is able to break the Flexible Square Diffie-Hellman assumption with probability ε/ℓ. □

The proofs of anonymity and security against framing attacks are identical to those of the first scheme and omitted here.

C.4 Further Reducing the Number of Assumptions

We note that, using the technique of Malkin, Teranishi, Vahlis and Yung [48], it is possible to replace the SDH assumption by the standard Diffie-Hellman assumption in the proof of security against framing attack. To this end, we must introduce a Waters-like [60] number theoretic hash function (described by $O(\lambda)$ group elements) in the group public key in order to have a message-dependent Groth-Sahai CRS. Namely, all proofs of the signature are generated w.r.t. a Groth-Sahai CRS $(\vec{f}_1, \vec{f}_2, \vec{f}_{\mathsf{VK}})$, where $\vec{f}_{\mathsf{VK}}$ is obtained by "hashing" the verification key of a one-time signature. In order to secure the scheme against framing attacks, each group signature should prove knowledge of a value (such as $g^{1/x}$, where $x = \log_g(X)$) that only the signer knows. Finally, all non-interactive proofs should be signed along with the actual message using the private key SK of the one-time signature[4].

The details are omitted here but it is not hard to see that a successful framing attack would imply a PPT algorithm to compute $g^{1/x}$ given $X = g^x$, which is equivalent to solving the Diffie-Hellman problem. Eventually, we only need the q-SFP assumption, the FSDH assumption and the DLIN assumption to prove the security of the scheme. In the resulting group signature, the group public key is larger and comprises $O(\lambda + \log^2 N)$ group elements.

[4] The reason why $\vec{f}_{\mathsf{VK}}$ is not directly derived from M is that we need to prevent Groth-Sahai proofs from being publicly randomized in order to achieve anonymity in the CCA2 sense: as noted in [35], signatures should not be re-randomizable in order to attain anonymity in the strongest sense.