Scalable Group Signatures with 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, simultaneously supporting accountability and anonymity. They allow users to anonymously sign messages on behalf of a group they are members of. The recent years saw the appearance of several constructions with security proofs in the standard model (*i.e.*, without appealing to the random oracle heuristic). For a digital signature scheme to be adopted, an efficient revocation scheme (as in regular PKI) is absolutely necessary. Despite over a decade of extensive research, membership revocation remains a non-trivial problem in group signatures: all existing solutions are not truly scalable due to either high overhead (e.g., large group public key size), or limiting operational requirement (the need for all users to follow the system's entire history). In the standard model, the situation is even worse as many existing solutions are not readily adaptable. To fill this gap and tackle this challenge, we describe a new revocation approach based, perhaps somewhat unexpectedly, on the Naor-Naor-Lotspiech framework which was introduced for a different problem (namely, that of broadcast encryption). Our mechanism yields efficient and scalable revocable group signatures in the standard model. In particular, the size of signatures and the verification cost are independent of the number of revocations and the maximal cardinality N of the group while other complexities are at most polylogarithmic in N. Moreover, the schemes are history-independent: unrevoked group members do not have to update their keys when a revocation occurs.

Keywords. Group signatures, revocation, standard model, efficiency.

1 Introduction

As suggested by Chaum and van Heyst in 1991 [31], *group signatures* allow members of a group to anonymously sign messages on behalf of a population group members managed by a group authority. Using some trapdoor information, a tracing authority must be able to "open" signatures and identify the signer. A complex problem in group signatures is the revocation of members whose signing capability should be disabled (either because they misbehaved or they intentionally leave the group).

⋆ This author was supported by the Belgian Fund for Scientific Research (F.R.S.-F.N.R.S.) via a "Chargé de recherches" fellowship.

⋆⋆ Supported by the IUAP B-Crypt Project and the Walloon Region Camus Project.

1.1 Related Work

Group signatures without revocation. The first efficient and provably coalition-resistant group signature was described by Ateniese, Camenisch, Joye and Tsudik in 2000 [7]. At that time, the security of group signatures was not totally understood and proper security definitions were given later on by Bellare, Micciancio and Warinschi [9] (BMW) whose model captures all the requirements of group signatures in three properties. In (a relaxation of) this model, Boneh, Boyen and Shacham [16] obtained a construction in the random oracle model [10] with signatures shorter than 200 bytes [13].

In the BMW model, the population of users is frozen after the setup phase beyond which no new member can be added. Dynamic group signatures were independently formalized by Kiayias and Yung [42] and Bellare-Shi-Zhang [11]. In these models, pairing-based schemes with relatively short signatures were put forth in [50, 32]. Ateniese *et al.* [6] also gave a construction without random oracles using interactive assumptions. In the BMW model [9], Boyen and Waters independently came up with a different standard model proposal [19] using more classical assumptions and they subsequently refined their scheme [20] to obtain constant-size signatures. In the dynamic model [11], Groth [37] described a system with constant-size signatures without random oracles but this scheme was rather a feasibility result than an efficient construction. Later on, Groth gave [38] a fairly efficient realization – with signatures consisting of about 50 group elements – in the standard model with the strongest anonymity level.

Revocation. In group signatures, membership revocation has received much attention in the last decade [21, 8, 28, 18] since revocation is central to digital signature schemes. One simple solution is to generate a new group public key and deliver a new signing key to each unrevoked member. However, in large groups, it may be inconvenient to change the public key and send a new secret to signers after they joined the group. An alternative approach taken by Bresson and Stern [21] is to have the signer prove that his membership certificate does not appear in a public list or revoked certificates. Unfortunately, the signer's workload and the size of signatures grow with the number of expelled users.

Song [51] presented an approach handling revocation in forward-secure group signatures. However, verification takes linear time in the number of revocations.

Using accumulators[1] [12], Camenisch and Lysyanskaya [28] proposed a method (followed by [55, 26]) to revoke users in the ACJT group signature [7] while keeping $O(1)$ costs for signing and verifying. While elegant, this approach is history-dependent and requires users to keep track of all changes in the population of the group: at each modification of the accumulator value, unrevoked users need to update their membership certificates before signing new messages, which may require up to $O(r)$ exponentiations if r is the number of revoked users.

Brickell [22] suggested the notion of *verifier-local revocation* group signatures, which was formalized by Boneh and Shacham [18] and further studied in [47,

[1] An accumulator allows hashing a set of values into a short string of constant size while allowing to efficiently prove that a specific value was accumulated.

56, 45]. In their systems, revocation messages are only sent to verifiers (making the signing algorithm independent of the number of revocations). The group manager maintains a revocation list (RL) which is used by verifiers to make sure that signatures were not generated by a revoked member. The RL contains a token for each revoked user and the verification algorithm has to verify signatures w.r.t. each token (a similar revocation mechanism is used in [23]). As a result, the verification cost is inevitably linear in the number of expelled users.

More recently, Nakanishi, Fuji, Hira and Funabiki [46] described a construction with constant complexities for signing/verifying and where group members never have to update their credentials. On the other hand, their proposal has the disadvantage of linear-size group public keys (in the maximal number N of users), although a tweak allows reducing the size to $O(N^{1/2})$.

In the context of anonymous credentials, Tsang *et al.* [53, 54] showed how to blacklist users without compromising their anonymity or involving a trusted third party. Their protocols either have linear proving complexity in the number of revocations or rely on accumulators (which may be problematic for our purposes). Camenisch, Kohlweiss and Soriente [27] handle revocations by periodically updating users credentials in which a specific attribute indicates a validity period. While useful in certain applications of anonymous credentials, in group signatures, their technique would place quite a burden on the group manager who would have to generate updates for each unrevoked individual credential.

1.2 Our Contribution

For the time being and despite years of research efforts, group signatures in the standard model have no revocation mechanism allowing for scalable (*i.e.*, constant or polylogarithmic) verification time without dramatically degrading the efficiency in other metrics and without being history-dependent. In pairing-based group signatures, accumulator-based approaches are unlikely to result in solutions supporting very large groups. The reason is that, in known pairing-based accumulators [49, 26], public keys have linear size in the maximal number of accumulated values (unless one sacrifices the constant size of proofs of non-membership as in [5]), which would result in linear-size group public keys in straightforward implementations. Recently [34], Fan *et al.* suggested a different way to use the accumulator of [26] and announced constant-size group public keys but their scheme still requires the group manager to publicize $O(N)$ values at each revocation. In a revocation mechanism along the lines of [28], Boneh, Boyen and Shacham [16] managed to avoid linear dependencies. However, their technique seems hard to combine[2] with Groth-Sahai proofs [39] so as to work in the standard model. In addition, we would like to save unrevoked users from

[2] In [16], 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 private key of the group manager, and the revocation mechanism 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 [20]), which is no longer compatible with the revocation technique.

having to update their keys after each revocation. To this end, it seems possible to adapt the approach of [46] in the standard model. However, merely replacing sigma-protocols by Groth-Sahai proofs in the scheme of [46] would result in group public keys of size $O(N^{1/2})$ in the best case.

In this paper, we describe a novel and scalable revocation technique that interacts nicely with Groth-Sahai proofs and gives constructions in the standard model with $O(1)$ verification cost and at most polylogarithmic complexity in other metrics. Our approach bears similarities with the one of Nakanishi *et al.* [46] in that it does not require users to update their membership certificates at any time but, unlike [46], our group public key size is either $O(\log N)$ or constant. Like the scheme of [46], our main system uses revocation lists (RLs) of size $O(r)$ – which is in line with RLs of standard PKIs – and we emphasize that these are *not* part of the group public key: verifiers only need to know the number of the latest revocation epoch and they do not have to read RLs entirely.

To obtain our constructions, we turn to the area of broadcast encryption and build on the Subset Cover framework of Naor, Naor and Lotspiech [48] (NNL). In a nutshell, the idea is to use the NNL ciphertext as a revocation list and have non-revoked signers prove their ability to decrypt in order to convince verifiers that they are not revoked. In its public-key variant, due to Dodis and Fazio [33], the Subset Cover framework relies on hierarchical identity-based encryption (HIBE) [41, 36] and each NNL ciphertext consists of several HIBE encryptions. To anonymously sign a message, we let group members commit to the specific HIBE ciphertext that they can decrypt (which gives constant-size signatures since only one ciphertext is committed to), and provide a non-interactive proof that: (i) they hold a private key which decrypts the committed HIBE ciphertext. (ii) The latter belongs to the revocation list.

By applying this approach to the Subset Difference (SD) method [48], we obtain a scheme with $O(1)$-size signatures, $O(\log N)$-size group public keys, membership certificates of size $O(\log^3 N)$ and revocation lists of size $O(r)$. The Layered Subset Difference method [40] can be used in the same way to obtain membership certificates of size $O(\log^{2.5} N)$. Using the Complete Subtree method, we obtain a tradeoff with $O(r \cdot \log N)$ revocation lists, log-size membership certificates and constant-size group public keys.

A natural question is whether our SD-based revocable group signatures can generically use any HIBE scheme. The answer is negative as the Boneh-Boyen-Goh (BBG) construction [15] is currently the only suitable candidate. Indeed, for anonymity reasons, ciphertexts should be of constant size and our security proof requires the HIBE system to satisfy a new and non-standard security property which is met by [15]. As we will see, the proof can hardly rely on the standard security notion for HIBE schemes [36].

We note that the new revocation mechanism can find applications in contexts other than group signatures. For example, it seems that it can be used in the oblivious transfer with access control protocol of [25], which also uses the technique of Nakanishi *et al.* [46] to revoke credentials.

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 $e(g,h) \neq 1_{\mathbb{G}_T}$ whenever $g,h \neq 1_{\mathbb{G}}$. We assume the hardness of several problems.

Definition 1 ([16]). *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 $a,b,c,d \overset{R}{\leftarrow} \mathbb{Z}_p^*$, $z \overset{R}{\leftarrow} \mathbb{Z}_p^*$.*

Definition 2 ([13]). *The* **q-Strong Diffie-Hellman problem** *(q-SDH) in $\mathbb{G}$ is, given $(g, g^a, \ldots, g^{(a^q)})$, for some $g \overset{R}{\leftarrow} \mathbb{G}$ and $a \overset{R}{\leftarrow} \mathbb{Z}_p$, to find a pair $(g^{1/(a+s)}, s) \in \mathbb{G} \times \mathbb{Z}_p$.*

We appeal to yet another "q-type" assumption introduced by Abe *et al.* [2].

Definition 3 ([2]). *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*

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

to find a new tuple $(z^\star, r^\star, s^\star, t^\star, u^\star, v^\star, w^\star) \in \mathbb{G}^7$ satisfying the above relation and such that $z^\star \neq 1_{\mathbb{G}}$ and $z^\star \neq z_j$ for $j \in \{1, \ldots, q\}$.

2.2 Groth-Sahai Proof Systems

In the following notations, for equal-dimension vectors or matrices A and B containing group elements, $A \odot B$ stands for their entry-wise product.

In their instantiations based on the DLIN assumption, the Groth-Sahai (GS) techniques [39] make use of prime order groups and a common reference string comprising 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 an element $X \in \mathbb{G}$, one sets $\vec{C} = (1,1,X) \odot \vec{f_1}^{\,r} \odot \vec{f_2}^{\,s} \odot \vec{f_3}^{\,t}$ with $r,s,t \overset{R}{\leftarrow} \mathbb{Z}_p^*$. When the CRS is configured to give perfectly sound proofs, we have $\vec{f_3} = \vec{f_1}^{\,\xi_1} \odot \vec{f_2}^{\,\xi_2}$ where $\xi_1, \xi_2 \in \mathbb{Z}_p^*$. Commitments to group elements $\vec{C} = (f_1^{r+\xi_1 t}, f_2^{s+\xi_2 t}, X \cdot g^{r+s+t(\xi_1+\xi_2)})$ are then Boneh-Boyen-Shacham (BBS) ciphertexts [16] that can be decrypted 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 a scalar $x \in \mathbb{Z}_p$, one computes $\vec{C} = \vec{\varphi}^{\,x} \odot \vec{f_1}^{\,r} \odot \vec{f_2}^{\,s}$, where $r,s \overset{R}{\leftarrow} \mathbb{Z}_p^*$, using a CRS comprising vectors $\vec{\varphi}, \vec{f_1}, \vec{f_2}$. In the soundness setting, $\vec{\varphi}, \vec{f_1}, \vec{f_2}$ are linearly independent (typically $\vec{\varphi} = \vec{f_3} \odot (1,1,g)$ where $\vec{f_3} = \vec{f_1}^{\,\xi_1} \odot \vec{f_2}^{\,\xi_2}$) whereas, in the WI setting, choosing $\vec{\varphi} = \vec{f_1}^{\,\xi_1} \odot \vec{f_2}^{\,\xi_2}$ gives a perfectly hiding commitment since $\vec{C}$ is always a BBS encryption of $1_{\mathbb{G}}$, no matter which exponent x is committed to.

To prove that committed variables satisfy a set of relations, the prover computes one commitment per variable and one proof element (made of a constant number of group elements) per relation.

Such proofs are available for pairing-product equations, which are of the type

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

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

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

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

In pairing-product equations, proofs for quadratic equations require 9 group elements whereas linear equations (*i.e.*, where $a_{ij} = 0$ for all i, j in equation (1)) only take 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 makes it is possible to simulate proofs without knowing witnesses and simulated proofs have the same distribution as real proofs.

2.3 Structure-Preserving Signatures

Several applications (see [2, 3, 35, 30, 4] for examples) require to sign groups elements while preserving the feasibility of efficiently proving that a committed signature is valid for a committed group element.

In [2, 3], Abe, Haralambiev and Ohkubo showed how to conveniently sign n group elements at once using signatures consisting of $O(1)$ group elements. Their scheme (which is referred to as the AHO signature in the paper) makes use of bilinear groups of prime order. In the context of symmetric pairings, the description below 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 that can be signed altogether, choose generators $G_r, H_r \overset{R}{\leftarrow} \mathbb{G}$. Pick $\gamma_z, \delta_z \overset{R}{\leftarrow} \mathbb{Z}_p$ and $\gamma_i, \delta_i \overset{R}{\leftarrow} \mathbb{Z}_p$, 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 $\alpha_a, \alpha_b \overset{R}{\leftarrow} \mathbb{Z}_p$ and define $A = e(G_r, g^{\alpha_a})$ and $B = e(H_r, g^{\alpha_b})$. The public key is

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

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

Sign$(sk, (M_1, \ldots, M_n))$**:** to sign a vector $(M_1, \ldots, M_n) \in \mathbb{G}^n$ using the private key $sk = (\alpha_a, \alpha_b, \gamma_z, \delta_z, \{\gamma_i, \delta_i\}_{i=1}^n)$, choose $\zeta, \rho, \tau, \nu, \omega \stackrel{R}{\leftarrow} \mathbb{Z}_p$ and compute $\theta_1 = g^\zeta$ as well as

$$\theta_2 = g^{\rho - \gamma_z \zeta} \cdot \prod_{i=1}^n M_i^{-\gamma_i}, \qquad \theta_3 = G_r^\tau, \qquad \theta_4 = g^{(\alpha_a - \rho)/\tau},$$

$$\theta_5 = g^{\nu - \delta_z \zeta} \cdot \prod_{i=1}^n M_i^{-\delta_i}, \qquad \theta_6 = H_r^\omega, \qquad \theta_7 = g^{(\alpha_b - \nu)/\omega},$$

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:

$$A = e(G_z, \theta_1) \cdot e(G_r, \theta_2) \cdot e(\theta_3, \theta_4) \cdot \prod_{i=1}^n e(G_i, M_i), \tag{3}$$

$$B = e(H_z, \theta_1) \cdot e(H_r, \theta_5) \cdot e(\theta_6, \theta_7) \cdot \prod_{i=1}^n e(H_i, M_i). \tag{4}$$

The scheme was proved [2, 3] existentially unforgeable under chosen-message attacks under the q-SFP assumption, where q is the number of signing queries.

Abe *et al.* [2, 3] also showed that 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 $\theta_1' = \theta_1$ while $\{\theta_i'\}_{i=2}^7$ are uniformly distributed among the values satisfying the equalities $e(G_r, \theta_2') \cdot e(\theta_3', \theta_4') = e(G_r, \theta_2) \cdot e(\theta_3, \theta_4)$ 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 $(M_1, \ldots, M_n)$ and the rest of the signature. This implies that, in anonymity-related protocols, re-randomized $\{\theta_i'\}_{i \in \{3,4,6,7\}}$ can be safely revealed as long as $(M_1, \ldots, M_n)$ and $\{\theta_i'\}_{i \in \{1,2,5\}}$ are given in committed form.

In [4], Abe, Groth, Haralambiev and Ohkubo described a more efficient structure-preserving signature based on interactive assumptions. Here, we use the scheme of [2, 3] so as to rely on non-interactive assumptions.

2.4 The NNL Framework for Broadcast Encryption

The Subset Cover framework [48] considers secret-key broadcast encryption schemes with $N = 2^\ell$ registered receivers. Each one of them is associated with a leaf of a complete binary tree T of height ℓ and each tree 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 idea of the framework 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 partition $\mathcal{N} \backslash \mathcal{R}$ and the distribution of keys to users, different instantiations and tradeoffs are possible.

The Complete Subtree Method. In this technique, each subset S_i consists of the leaves of a complete subtree rooted at some node x_i of T. Upon registration, each user obtains secret keys for all nodes on the path connecting his leaf to the root of T (and thus $O(\ell)$ keys overall). By doing so, users in $\mathcal{N}\backslash\mathcal{R}$ can decrypt the content if the latter is enciphered using symmetric keys $K_1,\ldots,K_m$ corresponding to the roots of subtrees $S_1,\ldots,S_m$. As showed in [48], the CS partitioning method entails at most $m \leq r\cdot\log(N/r)$ subsets, where $r = |\mathcal{R}|$. Each transmission requires to send $O(r\cdot\log N)$ symmetric encryptions while, at each user, the storage complexity is $O(\log N)$.

As noted in [48, 33], a single-level identity-based encryption scheme allows implementing a public-key variant of the CS method. The master public key of the IBE scheme forms the public key of the broadcast encryption system, which allows for public keys of size $O(1)$ (instead of $O(N)$ in instantiations using ordinary public-key encryption). When users join the system, they obtain $O(\ell)$ IBE private keys (in place of symmetric keys) associated with the "identities" of nodes on the path between their leaf and the root.

The Subset Difference Method. The SD method reduces the transmission cost to $O(r)$ at the expense of increased storage requirements. For each node $x_j \in \mathsf{T}$, we call T_{x_j} the subtree rooted at x_j. The set $\mathcal{N}\backslash\mathcal{R}$ is now divided 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 thus belongs to much more generic subsets than in the CS method and this allows reducing the maximal number of subsets to $m = 2r-1$ (see [48] for a proof of this bound).

A more complex key distribution is necessary here. 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}. At the same time, $P_{x_{k_i},x_{u_l}}$ should be hard to compute without a proto-key $P_{x_{k_i},x_{u_i}}$ for an ancestor x_{u_i} of x_{u_l}. The key distribution phase then proceeds as follows. Let user i be assigned a leaf v_i and let $\epsilon = x_0, x_1, \ldots, x_\ell = v_i$ denote the path from the root ϵ to v_i. For each subtree T_{x_j} (with $j \in \{1,\ldots,\ell\}$), if copath_{x_j} denotes the set of all siblings of nodes on the path from x_j to v_i, user i must obtain proto-keys $P_{x_j,w}$ for each node $w \in \mathsf{copath}_{x_j}$ because he belongs to the generic subset whose primary root is x_j and whose secondary root is w. By storing $O(\ell^2)$ proto-keys (*i.e.*, $O(\ell)$ for each subtree T_{x_j}), users will be able to derive keys for all generic subsets they belong to.

In [33], Dodis and Fazio extended the SD method to the public-key setting using hierarchical identity-based encryption. In the tree, each node w at depth $\leq \ell$ has a label $\langle w\rangle$ which is defined by assigning the label ε to the root (at depth 0). The left and right children of w are then labeled with $\langle w\rangle||0$ and $\langle w\rangle||1$, respectively. For each subset S_{k_i,u_i} of $\mathcal{N}\backslash\mathcal{R}$, the sender considers the primary and secondary roots x_{k_i}, x_{u_i} and parses the label $\langle x_{u_i}\rangle$ as $\langle x_{k_i}\rangle||u_{i,\ell_{i,1}}\ldots u_{i,\ell_{i,2}}$, with $u_{i,j} \in \{0,1\}$ for each $j \in \{\ell_{i,1},\ldots,\ell_{i,2}\}$. Then, he computes a HIBE ciphertext for the hierarchical identity $(\langle x_{k_i}\rangle, u_{i,\ell_{i,1}},\ldots,u_{i,\ell_{i,2}})$ at level $\ell_{i,2}-\ell_{i,1}+2$. Upon registration, if $\epsilon = x_0, \ldots, x_\ell = v_i$ denotes the path from the root to his leaf v_i, for each subtree T_{x_j}, user i receives exactly one HIBE private key for each $w \in \mathsf{copath}_{x_j}$: namely, for each $w \in \mathsf{copath}_{x_j}$, there exist $\ell_1, \ell_2 \in \{1, \ldots, \ell\}$ such that $\langle w \rangle = \langle x_j \rangle || w_{\ell_1} \ldots w_{\ell_2}$ with $w_j \in \{0, 1\}$ for all $j \in \{\ell_1, \ldots, \ell_2\}$ and user i obtains a HIBE private key for the hierarchical identity $(\langle x_j \rangle, w_{\ell_1}, \ldots, w_{\ell_2})$. By construction, this key will allow user i to decrypt any HIBE ciphertext encrypted for a subset whose primary root is x_j and whose secondary root is a descendant of w. Overall, each user thus has to store $O(\log^2 N)$ HIBE private keys.

2.5 Revocable Group Signatures

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 [46] but they build on those defined by Kiayias and Yung [42]. Like the Bellare-Shi-Zhang model [11], the latter assumes an interactive join protocol between the group manager and the user. This protocol provides the user with a membership certificate and a membership secret. Such protocols may consist of several rounds of interaction.

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 members $N \in \mathbb{N}$, this algorithm (which is run by a 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 $\mathcal{S}_{\mathsf{OA}}$. $\mathcal{S}_{\mathsf{GM}}$ and $\mathcal{S}_{\mathsf{OA}}$ are given to the appropriate authority while $\mathcal{Y}$ is publicized. The algorithm initializes a public state St containing a set data structure $St_{users} = \emptyset$ and a string structure $St_{\mathsf{trans}} = \epsilon$.

Join: is an interactive protocol between the group manager GM and a user $\mathcal{U}_i$ where the latter becomes a group member. 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 $[\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y}), \mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})]$, terminates with user $\mathcal{U}_i$ obtaining a membership secret sec_i, that no one else knows, and a membership certificate cert_i. If the protocol successfully terminates, 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 randomized) 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, that are formally defined in the full version of the paper. 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* mandates that, even if the whole system colludes against a user, that user will not bear responsibility for messages that he did not sign. Finally, the notion of *anonymity* is also defined (in the presence of a signature opening oracle) as in the models of [11, 42].

3 A Revocable Group Signature Based on the Subset Difference Method

The idea is to turn the NNL global ciphertext into a revocation list in the group signature. Each member is assigned to a leaf of a binary tree of height ℓ and the outcome of the join protocol is the user obtaining a membership certificate that contains the same key material as in the public-key variant of the SD method (*i.e.*, $O(\ell^2)$ HIBE private keys). To ensure traceability and non-frameability, these NNL private keys are linked to a group element X, that only the user knows the discrete logarithm of, by means of structure-preserving signatures.

At each revocation epoch t, the group manager generates an up-to-date revocation list RL_t consisting of $O(r)$ HIBE ciphertexts, each of which is signed using a structure-preserving signature. When it comes to sign a message, the user $\mathcal{U}_i$ proves that he is not revoked by providing evidence that he is capable of decrypting one of the HIBE ciphertexts in RL_t. To this end, $\mathcal{U}_i$ commits to that HIBE ciphertext C_l and proves that he holds a key that decrypts C_l. To convince the verifier that C_l belongs to RL_t, he proves knowledge of a signature on the committed HIBE ciphertext C_l (this technique is borrowed from the set membership proofs of [52, 24]). Of course, to preserve the anonymity of signers, we need a HIBE scheme with constant-size ciphertexts (otherwise, the length of the committed ciphertext could betray the signer's location in the tree), which is why the Boneh-Boyen-Goh construction [15] is the ideal candidate.

The scheme is made anonymous and non-frameable using the same techniques as Groth [38] in steps 4-6 of the signing algorithm. As for the security against misidentification attacks, we cannot prove it by relying on the standard collusion-resistance (captured by the definition of [36]) of the HIBE scheme. In the proof of security against misidentification attacks, the problem appears in the treatment of forgeries that open to a revoked user: while this user cannot have obtained a private key that decrypts the committed HIBE ciphertext of the forgery (because he is revoked), unrevoked adversarially-controlled users can. To solve this problem, we need to rest on a non-standard security property (formally defined in the full version of the paper) called "key-robustness". This notion asks that, given a private key generated for some hierarchical identity using specific random coins, it be infeasible to compute the private key of a different identity for the *same random coins* and even *knowing* the *master secret key* of the HIBE scheme. While unusual, this property can be proved (as shown in the full version of the paper) under the Diffie-Hellman assumption for the BBG construction.

Perhaps surprisingly, even though we rely on the BBG HIBE, we do not need its underlying q-type assumption [15]. The reason is that the master secret key of the scheme is unnecessary here as its role is taken over by the private key of a structure-preserving signature. In the ordinary BBG system, private keys contain components of the form $(g_2^{\alpha} \cdot F(\mathsf{ID})^r, g^r)$, for some $r \in \mathbb{Z}_p$, where g_2^{α} is the master secret key and $F(\mathsf{ID})$ is a function of the hierarchical identity. In the join protocol, the master key g_2^{α} disappears: the user obtains a private key of the form $(F(\mathsf{ID})^r, g^r)$ and an AHO signature is used to bind the user's membership public key X to g^r. The latter can be thought of as a public key for a one-time variant of the Boneh-Lynn-Shacham signature [17]. The underlying one-time private key $r \in \mathbb{Z}_p$ is used to compute $F(\mathsf{ID})^r$ as well as a number of delegation components allowing to derive signatures for messages that ID is a prefix of (somewhat in the fashion of wildcard signatures [1][Section 6]).

3.1 Construction

As in Section 2.4, $\langle x \rangle$ denotes the label of node $x \in \mathsf{T}$ and, for any sub-tree T_{x_j} rooted at x_j and any leaf v_i of T_{x_j}, copath_{x_j} denotes the set of all siblings of nodes on the path from x_j to v_i, not counting x_j itself.

As is standard in group signatures, the description below assumes 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 $\lambda \in \mathbb{N}$ and the permitted number of users $N = 2^{\ell}$,

1. Choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^{\lambda}$, with $g \stackrel{R}{\leftarrow} \mathbb{G}$.
2. 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 to sign messages of two group elements. These pairs consist of

$$pk_{\mathsf{AHO}}^{(d)} = \Big(G_r^{(d)},\ H_r^{(d)},\ G_z^{(d)} = G_r^{\gamma_z^{(d)}},\ H_z^{(d)} = H_r^{\delta_z^{(d)}},$$
$$\{G_i^{(d)} = G_r^{\gamma_i^{(d)}}, H_i^{(d)} = H_r^{\delta_i^{(d)}}\}_{i=1}^{2},\ A^{(d)},\ B^{(d)}\Big)$$

and $sk_{\mathsf{AHO}}^{(d)} = \big(\alpha_a^{(d)}, \alpha_b^{(d)}, \gamma_z^{(d)}, \delta_z^{(d)}, \{\gamma_i^{(d)}, \delta_i^{(d)}\}_{i=1}^{2}\big)$, where $d \in \{0, 1\}$.

3. 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 $f_1 = g^{\beta_1}, f_2 = g^{\beta_2} \overset{R}{\leftarrow} \mathbb{G}$ and $\beta_1, \beta_2, \xi_1, \xi_2 \overset{R}{\leftarrow} \mathbb{Z}_p^*$.
4. Choose $(U, V) \overset{R}{\leftarrow} \mathbb{G}^2$ that, together with f_1, f_2, g, will form a public encryption key.
5. Generate a master public key mpk_{BBG} for the Boneh-Boyen-Goh HIBE. Such a public key consists[3] of $mpk_{\mathsf{BBG}} = \big(\ \{h_i\}_{i=0}^{\ell}\big)$, where $\ell = \log_2(N)$, and no master secret key is needed.
6. Select an injective encoding[4] function $\mathcal{H} : \{0,1\}^{\leq \ell} \rightarrow \mathbb{Z}_p^*$ and 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

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

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

1. $\mathsf{J}_{\mathsf{user}}(\lambda, \mathcal{Y})$ computes $X = g^x$, for a randomly chosen $x \overset{R}{\leftarrow} \mathbb{Z}_p$, and sends it to $\mathsf{J}_{\mathsf{GM}}(\lambda, St, \mathcal{Y}, \mathcal{S}_{\mathsf{GM}})$. If the value X already appears in some entry $\mathsf{transcript}_j$ of the database St_{trans}, J_{GM} aborts and returns $\perp$ to $\mathsf{J}_{\mathsf{user}}$.
2. J_{GM} assigns to $\mathcal{U}_i$ an available leaf v_i of label $\langle v_i \rangle = v_{i,1} \dots v_{i,\ell} \in \{0,1\}^{\ell}$ in the tree T. Let $x_0 = \epsilon$, x_1, $\dots$, $x_{\ell-1}$, $x_\ell = v_i$ be the path from v_i to the root ϵ of T. For $j = 0$ to ℓ, J_{GM} does the following.

   a. Consider the sub-tree T_{x_j} rooted at node x_j. Let copath_{x_j} be the co-path from x_j to v_i.

   b. For each node $w \in \mathsf{copath}_{x_j}$, since x_j is an ancestor of w, $\langle x_j \rangle$ is a prefix of $\langle w \rangle$ and we denote by $w_{\ell_1} \dots w_{\ell_2} \in \{0,1\}^{\ell_2 - \ell_1 + 1}$, for some $\ell_1 \leq \ell_2 \leq \ell$, the suffix of $\langle w \rangle$ coming right after $\langle x_j \rangle$.

   b.1 Choose a random $r \overset{R}{\leftarrow} \mathbb{Z}_p$ and compute a HIBE private key

   $$\begin{aligned} d_w &= (D_{w,1}, D_{w,2}, K_{w,\ell_2-\ell_1+3}, \dots, K_{w,\ell}) \\ &= \Big(\big(h_0 \cdot h_1^{\mathcal{H}(\langle x_j \rangle)} \cdot h_2^{\mathcal{H}(w_{\ell_1})} \cdots h_{\ell_2-\ell_1+2}^{\mathcal{H}(w_{\ell_2})}\big)^r,\ g^r,\ h_{\ell_2-\ell_1+3}^r, \dots,\ h_\ell^r\Big) \end{aligned}$$

   for the identity $(\mathcal{H}(\langle x_j \rangle), \mathcal{H}(w_{\ell_1}), \dots, \mathcal{H}(w_{\ell_2})) \in (\mathbb{Z}_p^*)^{\ell_2 - \ell_1 + 2}$.

   b.2 Using $sk_{\mathsf{AHO}}^{(0)}$, generate an AHO signature $\sigma_w = (\theta_{w,1}, \dots, \theta_{w,7})$ on $(X, D_{w,2}) \in \mathbb{G}^2$ so as to bind the HIBE private key d_w to the value X that identifies $\mathcal{U}_i$.

[3] In comparison with the original HIBE scheme where mpk_{BBG} includes $(g_1 = g^{\alpha}, g_2)$ and $msk_{\mathsf{BBG}} = g_2^{\alpha}$, the public elements g_1 and g_2 have disappeared.

[4] This encoding allows making sure that "identities" will be non-zero at each level. Since the set $\{0,1\}^{\leq \ell}$ is of cardinality $\sum_{i=0}^{\ell} 2^i = 2^{\ell+1} - 1 < p - 1$, such a function can be efficiently constructed without any intractability assumption.

3. J_{GM} sends $\langle v_i \rangle \in \{0,1\}^\ell$, and the HIBE private keys $\{\{d_w\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}$ to $\mathsf{J}_{\mathsf{user}}$ that verifies their validity. If these keys are all well-formed, $\mathsf{J}_{\mathsf{user}}$ acknowledges them by generating an ordinary digital signature $sig_i = \mathrm{Sign}_{\mathsf{usk}[i]}\big(X||\{\{d_w\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}\big)$ and sends it back to J_{GM}.
4. J_{GM} checks that $\mathrm{Verify}_{\mathsf{upk}[i]}\big(X||\{\{d_w\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}, sig_i\big) = 1$. If not, then J_{GM} aborts. Otherwise, J_{GM} returns the set of AHO signatures $\{\{\sigma_w\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}$ to $\mathsf{J}_{\mathsf{user}}$ and stores the entire conversation transcript $\mathsf{transcript}_i = (X, \{\{d_w, \sigma_w\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}, sig_i)$ in the database St_{trans}.
5. $\mathsf{J}_{\mathsf{user}}$ defines user $\mathcal{U}_i$'s membership certificate cert_i to be the tuple $\mathsf{cert}_i = \big(\langle v_i \rangle, \{\{d_w, \sigma_w\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}, X\big)$, where X will serve as the tag that identifies $\mathcal{U}_i$. The membership secret sec_i is defined to be $\mathsf{sec}_i = x$.

Revoke$(\mathcal{Y}, \mathcal{S}_{\mathsf{GM}}, t, \mathcal{R}_t)$**:** Parse $\mathcal{S}_{\mathsf{GM}}$ as $\mathcal{S}_{\mathsf{GM}} := \big(sk_{\mathsf{AHO}}^{(0)}, sk_{\mathsf{AHO}}^{(1)}\big)$. Using the SD covering algorithm, find a cover of the unrevoked user set $\{1,\ldots,N\}\backslash\mathcal{R}_t$ as the union of disjoint subsets $S_{k_1,u_1},\ldots,S_{k_m,u_m}$, with $m \leq 2\cdot|\mathcal{R}_t| - 1$. Then, for $i = 1$ to m, do the following.

a. Consider 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}. The label of x_{u_i} can be written $\langle x_{u_i}\rangle = \langle x_{k_i}\rangle || u_{i,\ell_{i,1}} \ldots u_{i,\ell_{i,2}}$ for some $\ell_{i,1} < \ell_{i,2} \leq \ell$ and where $u_{i,\kappa} \in \{0,1\}$ for each $\kappa \in \{\ell_{i,1},\ldots,\ell_{i,2}\}$. Then, compute an encoding of S_{k_i,u_i} as a group element

$$C_i = h_0 \cdot h_1^{\mathcal{H}(\langle x_{k_i}\rangle)} \cdot h_2^{\mathcal{H}(u_{i,\ell_{i,1}})} \cdots h_{\ell_{i,2}-\ell_{i,1}+2}^{\mathcal{H}(u_{i,\ell_{i,2}})},$$

which can be seen as a de-randomized HIBE ciphertext for the hierarchical identity $\big(\mathcal{H}(\langle x_{k_i}\rangle), \mathcal{H}(u_{i,\ell_{i,1}}),\ldots,\mathcal{H}(u_{i,\ell_{i,2}})\big) \in (\mathbb{Z}_p^*)^{\ell_{i,2}-\ell_{i,1}+2}$.

b. To authenticate the HIBE ciphertext C_i and bind it to the revocation epoch t, use $sk_{\mathsf{AHO}}^{(1)}$ to generate an AHO signature $\Theta_i = (\Theta_{i,1},\ldots,\Theta_{i,7}) \in \mathbb{G}^7$ on the pair $(C_i, g^t) \in \mathbb{G}^2$, where the epoch number t is interpreted as an element of $\mathbb{Z}_p$.

Return the revocation data RL_t which is defined to be

$$RL_t = \Big(t,\ \mathcal{R}_t,\ \{\langle x_{k_i}\rangle,\ \langle x_{u_i}\rangle,\ \big(C_i, \Theta_i = (\Theta_{i,1},\ldots,\Theta_{i,7})\big)\}_{i=1}^{m}\Big) \qquad (5)$$

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 $\big(\langle v_i\rangle, \{\{(d_w,\sigma_w)\}_{w\in\mathsf{copath}_{x_j}}\}_{j=0}^{\ell}, X\big)$ and sec_i as $x \in \mathbb{Z}_p$.

1. Using RL_t, determine the set S_{k_l,u_l}, with $l \in \{1,\ldots,m\}$, that contains the leaf v_i (this subset must exist since $i \notin \mathcal{R}_t$) and let x_{k_l} and x_{u_l} denote the primary and secondary roots of S_{k_l,u_l}. Since x_{k_l} is an ancestor of x_{u_l}, we can write $\langle x_{u_l}\rangle = \langle x_{k_l}\rangle || u_{l,\ell_1}\ldots u_{l,\ell_2}$, for some $\ell_1 < \ell_2 \leq \ell$ and with $u_{l,\kappa} \in \{0,1\}$ for each $\kappa \in \{\ell_1,\ldots,\ell_2\}$. The signer $\mathcal{U}_i$ computes a HIBE decryption key of the form

$$(D_{l,1}, D_{l,2}) = \Big(\big(h_0 \cdot h_1^{\mathcal{H}(\langle x_{k_l}\rangle)} \cdot h_2^{\mathcal{H}(u_{l,\ell_1})} \cdots h_{\ell_2-\ell_1+2}^{\mathcal{H}(u_{l,\ell_2})}\big)^r,\ g^r\Big). \qquad (6)$$

This is possible since, if we denote by $\langle x_{k,l} \rangle || u_{l,\ell_1} \dots u_{l,\ell'_1}$ the shortest prefix of $\langle x_{u_l} \rangle$ that is not a prefix of $\langle v_i \rangle$, the key material $\{d_w\}_{w \in \mathsf{copath}_{x_{k_l}}}$ corresponding to the sub-tree rooted at x_{k_l} contains a HIBE private key $d_w = (D_{w,1}, D_{w,2}, K_{w,\ell'_1-\ell_1+3}, \dots, K_{w,\ell})$ such that

$$d_w = \Big(\big(h_0 \cdot h_1^{\mathcal{H}(\langle x_{k_l} \rangle)} \cdot h_2^{\mathcal{H}(u_{l,\ell_1})} \cdots h_{\ell'_1-\ell_1+2}^{\mathcal{H}(u_{l,\ell'_1})}\big)^r,\ g^r,\ h^r_{\ell'_1-\ell_1+3}, \dots, h^r_\ell \Big),$$

which allows deriving a key of the form (6) such that $D_{l,2} = D_{w,2}$.

2. To prove his ability to "decrypt" C_l, user $\mathcal{U}_i$ first re-randomizes Θ_l as $\{\Theta'_{l,i}\}_{i=1}^7 \leftarrow \mathsf{ReRand}(pk^{(1)}_{\mathsf{AHO}}, \Theta_l)$. Then, he computes a Groth-Sahai commitment com_{C_l} to C_l as well as commitments $\{com_{\Theta'_{l,i}}\}_{i \in \{1,2,5\}}$ to $\{\Theta'_{l,i}\}_{i \in \{1,2,5\}}$. He generates a proof π_{C_l} that C_l is a certified HIBE ciphertext for epoch t: *i.e.*, π_{C_l} provides evidence that

$$\begin{aligned} A^{(1)} \cdot e(\Theta'_{l,3}, \Theta'_{l,4})^{-1} \ &\cdot\ e(G_2^{(1)}, g^t)^{-1} \\ &= e(G_z^{(1)}, \Theta'_{l,1}) \cdot e(G_r^{(1)}, \Theta'_{l,2}) \cdot e(G_1^{(1)}, C_l), \qquad (7) \\ B^{(1)} \cdot e(\Theta'_{l,6}, \Theta'_{l,7})^{-1} \ &\cdot\ e(H_2^{(1)}, g^t)^{-1} \\ &= e(H_z^{(1)}, \Theta'_{l,1}) \cdot e(H_r^{(1)}, \Theta'_{l,5}) \cdot e(H_1^{(1)}, C_l). \qquad (8) \end{aligned}$$

Then, $\mathcal{U}_i$ generates commitments $\{com_{D_{l,i}}\}_{i=1}^2$ to the HIBE key components $\{D_{l,i}\}_{i=1}^2$ derived at step 1 and computes a proof π_{D_l} that $e(D_{l,1}, g) = e(C_l, D_{l,2})$. The latter is quadratic and requires 9 group elements. Since $\{\Theta'_{l,i}\}_{i \in \{3,4,6,7\}}$ are constants, equations (7) are linear and require 3 elements each. So, π_{C_l} and π_{D_l} take 15 elements altogether.

3. Let $\sigma_l = (\theta_{l,1}, \dots, \theta_{l,7})$ be the AHO signature on $(X, D_{l,2})$. Compute $\{\theta'_{l,i}\}_{i=1}^7 \leftarrow \mathsf{ReRand}(pk^{(0)}_{\mathsf{AHO}}, \sigma_l)$ as well as commitments $\{com_{\theta'_{l,i}}\}_{i \in \{1,2,5\}}$ to $\{\theta'_{l,i}\}_{i \in \{1,2,5\}}$ and a commitment com_X to X. Then, generate a proof π_{σ_l} that committed variables satisfy the verification equations

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

Since these equations are linear, π_{σ_l} requires 6 group elements.

4. Using VK as a tag (we assume that it is first hashed onto $\mathbb{Z}_p$ in such a way that it can be interpreted as a $\mathbb{Z}_p$ element), compute a tag-based encryption [44] of X by drawing $z_1, z_2 \stackrel{R}{\leftarrow} \mathbb{Z}_p$ and setting

$$(\Psi_1, \Psi_2, \Psi_3, \Psi_4, \Psi_5) = \big(f_1^{z_1}, f_2^{z_2}, X \cdot g^{z_1+z_2}, (g^{\mathsf{VK}} \cdot U)^{z_1}, (g^{\mathsf{VK}} \cdot V)^{z_2}\big).$$

5. Generate a NIZK proof that $com_X = (1, 1, X) \cdot \vec{f_1}^{\,\phi_{X,1}} \cdot \vec{f_2}^{\,\phi_{X,2}} \cdot \vec{f_3}^{\,\phi_{X,3}}$ and (Ψ_1, Ψ_2, Ψ_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 $(f_1^{\phi_{X,1}} \cdot f_{3,1}^{\phi_{X,3}}, f_2^{\phi_{X,2}} \cdot f_{3,2}^{\phi_{X,3}}, X \cdot g^{\phi_{X,1}+\phi_{X,2}} \cdot f_{3,3}^{\phi_{X,3}})$, so that we have

$$com_X \odot (\Psi_1, \Psi_2, \Psi_3)^{-1} = \big(f_1^{\tau_1} \cdot f_{3,1}^{\tau_3},\ f_2^{\tau_2} \cdot f_{3,2}^{\tau_3},\ g^{\tau_1+\tau_2} \cdot f_{3,3}^{\tau_3}\big) \tag{9}$$

with $\tau_1 = \phi_{X,1} - z_1$, $\tau_2 = \phi_{X,2} - z_2$, $\tau_3 = \phi_{X,3}$. The signer $\mathcal{U}_i$ commits to $\tau_1, \tau_2, \tau_3 \in \mathbb{Z}_p$ (by computing $com_{\tau_j} = \vec{\varphi}^{\tau_j} \cdot \vec{f_1}^{\phi_{\tau_j,1}} \cdot \vec{f_2}^{\phi_{\tau_j,2}}$, for $j \in \{1,2,3\}$, using the vector $\vec{\varphi} = \vec{f_3} \cdot (1,1,g)$ and random $\{\phi_{\tau_j,1}, \phi_{\tau_j,2}\}_{j=1}^{3}$), and generates proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^{3}$ that τ_1, τ_2, τ_3 satisfy the three relations (9). Since these are linear equations, proofs $\{\pi_{eq\text{-}com,j}\}_{j=1}^{3}$ cost 2 elements each.

6. Compute $\sigma_{\mathsf{VK}} = g^{1/(x+\mathsf{VK})}$ and generate a commitment $com_{\sigma_{\mathsf{VK}}}$ to σ_{VK}. Then, generate a NIWI proof that committed variables σ_{VK} and X satisfy $e(\sigma_{\mathsf{VK}}, X \cdot g^{\mathsf{VK}}) = e(g,g)$. This relation is quadratic and costs 9 group elements to prove. We denote this proof by $\pi_{\sigma_{\mathsf{VK}}} = (\vec{\pi}_{\sigma_{\mathsf{VK}},1}, \vec{\pi}_{\sigma_{\mathsf{VK}},2}, \vec{\pi}_{\sigma_{\mathsf{VK}},3})$.
7. Compute $\sigma_{ots} = \mathcal{S}(\mathsf{SK}, (M, RL_t, \Psi_1, \Psi_2, \Psi_3, \Psi_4, \Psi_5, \Omega, \mathbf{com}, \mathbf{\Pi}))$, where we define $\Omega = \{\Theta'_{l,i}, \theta'_{l,i}\}_{i \in \{3,4,6,7\}}$, and

$$\begin{aligned}\mathbf{com} = \big(&com_{C_l}, \{com_{D_{l,i}}\}_{i=1}^{2}, com_X, \{com_{\Theta'_{l,i}}\}_{i\in\{1,2,5\}}, \\ &\qquad\qquad \{com_{\theta'_{l,i}}\}_{i\in\{1,2,5\}}, \{com_{\tau_i}\}_{i=1}^{3}, com_{\sigma_{\mathsf{VK}}}\big)\\ \mathbf{\Pi} = \big(&\pi_{C_l}, \pi_{D_l}, \pi_{\sigma_l}, \pi_{eq\text{-}com,1}, \pi_{eq\text{-}com,2}, \pi_{eq\text{-}com,3}, \pi_{\sigma_{\mathsf{VK}}}\big)\end{aligned}$$

Return the signature $\sigma = \big(\mathsf{VK}, \Psi_1, \Psi_2, \Psi_3, \Psi_4, \Psi_5, \Omega, \mathbf{com}, \mathbf{\Pi}, \sigma_{ots}\big)$.

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

1. If $\mathcal{V}(\mathsf{VK}, (\Psi_1, \Psi_2, \Psi_3, \Psi_4, \Psi_5, \Omega, \mathbf{com}, \mathbf{\Pi}), \sigma_{ots}) = 0$, return 0.
2. Return 0 if $e(\Psi_1, g^{\mathsf{VK}} \cdot U) \neq e(f_1, \Psi_4)$ or $e(\Psi_2, g^{\mathsf{VK}} \cdot V) \neq e(f_2, \Psi_5)$.
3. Return 1 if all proofs properly verify. Otherwise, return 0.

Open$(M, t, RL_t, \sigma, \mathcal{S}_{\mathsf{OA}}, \mathcal{Y}, St)$**:** given $\mathcal{S}_{\mathsf{OA}} = (\beta_1, \beta_2)$, parse the signature σ as above and return $\perp$ if $\mathsf{Verify}(\sigma, M, t, RL_t, \mathcal{Y}) = 0$. Otherwise, compute the value $\tilde{X} = \Psi_3 \cdot \Psi_1^{-1/\beta_1} \cdot \Psi_2^{-1/\beta_2}$. In the database of transcripts St_{trans}, find a record $\langle i, \mathsf{transcript}_i = (X, \{\{d_w, \sigma_w\}_{w \in \mathsf{copath}_{x_j}}\}_{j=0}^{\ell}, sig_i)\rangle$ such that $X = \tilde{X}$. If no such record exists in St_{trans}, return $\perp$. Otherwise, return i.

From an efficiency point of view, for each $i \in \{1 \ldots, m\}$, RL_t comprises 8 group elements plus the labels of nodes that identify S_{k_i,u_i}. If $\lambda_{\mathbb{G}}$ denotes the bitlength of a group element, the number of bits of RL_t is thus bounded by $2 \cdot |\mathcal{R}_t| \cdot (8 \cdot \lambda_{\mathbb{G}} + 2 \log N) < 2 \cdot |\mathcal{R}_t| \cdot (9\lambda_{\mathbb{G}})$ bits (as $\log N < \lambda_{\mathbb{G}}/2$ since $\lambda \leq \lambda_{\mathbb{G}}$ and N is polynomial). The size of revocation lists thus amounts to that of at most $18 \cdot |\mathcal{R}_t|$ group elements.

Users need $O(\log^3 N)$ group elements to store their membership certificate. As far as the size of signatures goes, $\mathbf{com}$ and $\mathbf{\Pi}$ require 42 and 36 group elements, respectively. If the one-time signature of [37] is used, σ consists of 96 group elements, which is less than twice the size of Groth's signatures [38]. At the 128-bit security level, a signature takes 6 kB.

Verifying signatures takes constant time. The cost of each signature generation is dominated by at most $\ell = \log N$ exponentiations to derive a HIBE private key at step 1. However, this step only has to be executed once per revocation epoch, at the first signature of that epoch.

The scheme is proved secure against misidentification attacks assuming the hardness of the q-SFP problem, where q is a polynomial function of $\ell = \log_2 N$, the number of adversarially-controlled users and the number of revocations. The security against framing attacks is proved under the SDH assumption and assuming that the one-time signature is strongly unforgeable. As for the anonymity property, we prove it under the DLIN assumption and assuming the strong unforgeability of the one-time signature. Due to space limitation, all security proofs are deferred to the full version of the paper.

References

1. M. Abdalla, E. Kiltz, G. Neven. Generalized Key Delegation for Hierarchical Identity-Based Encryption. In *ESORICS'07*, *LNCS* 4734, pp. 139–154. Springer, 2007.
2. M. Abe, K. Haralambiev, M. Ohkubo. Signing on Elements in Bilinear Groups for Modular Protocol Design. Cryptology ePrint Archive: Report 2010/133, 2010.
3. 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.
4. 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.
5. T. Acar, L. Nguyen. Revocation for Delegatable Anonymous Credentials. In *PKC'11*, *LNCS* 6571, pp. 423–440, 2011.
6. G. Ateniese, J. Camenisch, S. Hohenberger, B. de Medeiros. Practical group signatures without random oracles. Cryptology ePrint Archive: Report 2005/385, 2005.
7. 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.
8. G. Ateniese, D. Song, G. Tsudik. Quasi-Efficient Revocation in Group Signatures. In *Financial Cryptography'02*, *LNCS* 2357, pp. 183–197, 2002.
9. 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.
10. 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.
11. 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.
12. J. Benaloh, M. de Mare. One-Way Accumulators: A Decentralized Alternative to Digital Sinatures. In *Eurocrypt'93*, *LNCS* 4948, pp. 274–285, 1993.

13. D. Boneh, X. Boyen. Short Signatures Without Random Oracles. In *Eurocrypt'04*, *LNCS* 3027, pp. 56–73. Springer-Verlag, 2004.
14. D. Boneh, X. Boyen. Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In *Eurocrypt'04*, *LNCS* 3027, pp. 223–238, 2004.
15. D. Boneh, X. Boyen, E.-J. Goh. Hierarchical identity based encryption with constant size ciphertext. In *Eurocrypt'05*, *LNCS* 3494, pp. 440–456, 2005.
16. D. Boneh, X. Boyen, H. Shacham. Short Group Signatures. In *Crypto'04*, *LNCS* 3152, pp. 41–55. Springer, 2004.
17. D. Boneh, B. Lynn, H. Shacham. Short signatures from the Weil pairing. In *Asiacrypt'01*, *LNCS* 2248, pp. 514–532. Springer, 2001.
18. D. Boneh, H. Shacham. Group signatures with verifier-local revocation. In *ACM-CCS'04*, pp. 168–177. ACM Press, 2004.
19. X. Boyen, B. Waters. Compact Group Signatures Without Random Oracles. In *Eurocrypt'06*, *LNCS* 4004, pp. 427–444, Springer, 2006.
20. X. Boyen, B. Waters. Full-Domain Subgroup Hiding and Constant-Size Group Signatures. In *PKC'07*, *LNCS* 4450, pp. 1–15, 2007.
21. E. Bresson, J. Stern. Efficient Revocation in Group Signatures. In *PKC'01*, *LNCS* 1992, pp. 190–206, 2001.
22. E. Brickell. An efficient protocol for anonymously providing assurance of the container of the private key. Submission to the Trusted Computing Group. April, 2003.
23. E. Brickell, J. Camenisch, L. Chen. Direct Anonymous Attestation. In *ACM-CCS'04*, pp. 132–145, 2004.
24. J. Camenisch, R. Chaabouni, a. shelat. Efficient Protocols for Set Membership and Range Proofs. In *Asiacrypt'08*, *LNCS* 5350, pp. 234–252, Springer, 2008.
25. J. Camenisch, M. Dubovitskaya, G. Neven, G. Zaverucha. Oblivious Transfer with Hidden Access Control Policies. In *PKC'11*, *LNCS* 6571, pp. 192–209, 2011.
26. 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.
27. J. Camenisch, M. Kohlweiss, C. Soriente. Solving Revocation with Efficient Update of Anonymous Credentials. In *SCN'10*, *LNCS* 6280, pp. 454–471, 2010.
28. J. Camenisch, A. Lysyanskaya. Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In *Crypto'02*, *LNCS* 2442, pp. 61–76, Springer, 2002.
29. R. Canetti, S. Halevi, J. Katz. A Forward-Secure Public-Key Encryption Scheme. In *Eurocrypt'03*, *LNCS* 2656, pp. 254–271, 2003.
30. J. Cathalo, B. Libert, M. Yung. Group Encryption: Non-Interactive Realization in the Standard Model. In *Asiacrypt'09*, *LNCS* 5912, pp. 179–196, Springer, 2009.
31. D. Chaum, E. van Heyst. Group Signatures. In *Eurocrypt'91*, *LNCS* 547, pp. 257–265, Springer, 1991.
32. C. Delerablée, D. Pointcheval. Dynamic Fully Anonymous Short Group Signatures. In *Vietcrypt'06*, *LNCS* 4341, pp. 193–210, Springer, 2006.
33. Y. Dodis, N. Fazio. Public Key Broadcast Encryption for Stateless Receivers. In *Digital Rights Management (DRM'02)*, *LNCS* 2696, pp. 61–80, 2002.
34. C.-I. Fan, R.-H. Hsu, M. Manulis. Group Signature with Constant Revocation Costs for Signers and Verifiers. In *Cryptology and Network Security (CANS 2011)*, *LNCS* 7092, pp. 214–233, 2011.
35. G. Fuchsbauer. Automorphic Signatures in Bilinear Groups and an Application to Round-Optimal Blind Signatures. Cryptology ePrint Archive: Report 2009/320, 2009.

36. C. Gentry, A. Silverberg. Hierarchical ID-based cryptography. In *Asiacrypt'02*, *LNCS* 2501, Springer, 2002.
37. 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.
38. J. Groth. Fully anonymous group signatures without random oracles. In *Asiacrypt 2007*, *LNCS* 4833, pp. 164–180. Springer, 2007.
39. J. Groth, A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Eurocrypt'08*, LNCS 4965, pp. 415–432, 2008.
40. D. Halevy, A. Shamir. The LSD broadcast encryption scheme. In *Crypto'02*, *LNCS* 2442, pp. 47–60, Springer, 2002.
41. J. Horwitz, B. Lynn. Toward hierarchical identity-based encryption. In *Eurocrypt'02*, *LNCS* 2332, Springer, 2002.
42. 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.
43. A. Kiayias, M. Yung. Group signatures with efficient concurrent join. In *Eurocrypt'05*, *LNCS* 3494, pp. 198–214, 2005.
44. E. Kiltz. Chosen-ciphertext security from tag-based encryption. In *TCC'06*, *LNCS* 3876, pp. 581–600, 2006.
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. 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.
47. 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.
48. M. Naor, D. Naor, J. Lotspiech. Revocation and Tracing Schemes for Stateless Receivers. In *Crypto'01*, *LNCS* 2139, pp. 41–62, 2001.
49. L. Nguyen. Accumulators from Bilinear Pairings and Applications. In *CT-RSA'05*, *LNCS* 3376, pp. 275–292, 2005.
50. 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.
51. D. Song. Practical forward secure group signature schemes. In *ACM-CCS'01*, pp. 225–234, 2001.
52. I. Teranishi, K. Sako. k-Times Anonymous Authentication with a Constant Proving Cost. In *PKC'06*, *LNCS* 3958, pp. 525–542, Springer, 2006.
53. 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.
54. 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.
55. G. Tsudik, S. Xu. Accumulating Composites and Improved Group Signing. In *Asiacrypt'03*, *LNCS* 2894, pp. 269–286, 2003.
56. S. Zhou, D. Lin. Shorter Verifier-Local Revocation Group Signatures from Bilinear Maps. In *CANS'06*, *LNCS* 4301, pp. 126–143, Springer, 2006.