Zero-Knowledge Proof System for Fully Anonymous Attribute Based Group Signatures from Lattices with VLR

Maharage Nisansala Sevwandi Perera, Toru Nakamura, Masayuki Hashimoto,
and Hiroyuki Yokoyama

Adaptive Communications Research Laboratories,
Advanced Telecommunications Research Institute International (ATR),
Kyoto, Japan
{perera.nisansala, tr-nakamura, masayuki.hashimoto, hr-yokoyama}@atr.jp

Abstract. Signature schemes with Verifier-Local Revocation (VLR) fail to achieve stronger anonymity notion, full-anonymity. In full-anonymity, it is free to corrupt the secret signing keys. Secret signing keys of VLR schemes consist of tokens which can be used to identify the users. Thus VLR schemes restrict corrupting secret signing keys. VLR schemes can achieve full-anonymity by separating tokens from secret signing keys. However, separation of tokens gives space to signers to replace tokens with fake values. Generating signatures with fake tokens can be prevented with a suitable proof system. This paper proposes a new zero-knowledge protocol to support provers to convince verifiers, that attributes used for creating the signature are valid and have naive tokens. Moreover, this paper offers a new Attribute-Based Group Signature (ABGS) scheme, that uses the proposed protocol and achieves full anonymity.

Keywords: Attribute-Based Group Signatures · Verifier-Local Revocation · Zero-Knowledge Proof · Full Anonymity · Lattice-based Cryptography

1 Introduction

Attribute-Based Group Signatures (ABGS) allow a verifier to request a signature from a group who possesses specific attributes [14]. Thus, only a group member possessing required attributes can sign anonymously on behalf of the group. ABGS schemes belong to the family of Digital Signature (DS) schemes such as Group Signature (GS) schemes and Ring Signature (RS) schemes. ABGS scheme is a combination of Group Signature Schemes and Attribute-Based Signatures.

Group Signatures were first introduced by Chaum and Van Heyst [2], and since then, different lines of works were presented to achieve security and efficiency. However, due to the two characteristics; *Anonymity* and *Traceability* of naive group signature schemes, most of the researchers interested in applying Group signatures in real-life systems. The anonymity allows any group member to output a signature while hiding his identity among the group members. The traceability grants an authorized person to cancel the anonymity of a valid signature. Thus, group signature schemes produce signatures which are anonymous to the verifiers (outsiders) and known to the authorities.

Attribute-Based Signatures (ABS), which is a generalization of the digital signatures, allows a user to generate a signature over some specified attributes while being anonymous. In an ABS scheme, a user can generate a signature only if he possesses the attributes required in a given policy. Thus, the signer should possess the necessary attributes to create a signature, and the verifier may check whether the signature is generated by satisfying the policy requirements. The security of ABS ensures the privacy of the signer. Thus, the signer should not reveal any information related to the attributes. ABS schemes were first introduced by Maji et al. [21] in a preliminary version. Later, other ABS schemes [4,5,7,9,10,17,18,25] presented improvements like pairing efficiency, constant-size signatures, user-control linkability, and decentralized-traceability.

Dalia Khader proposed the first Attribute-Based Group Signature (ABGS) scheme [14]. In their scheme, the verifier can determine the role of the signer. Again, Dalia Khader presented another ABGS scheme with a revocation method [13]. However, both schemes are not secure under quantum attacks as they both were constructed using bilinear mappings. Recently, Kuchta et al. [15] and Zhang et al. [27] presented ABGS schemes from lattices. While Kuchta's work focuses on member registration, Zhang's work produces an ABGS scheme with revocation. In Zhang's scheme [27], a member revocation method called *Verifier-local Revocation* (*VLR*) is used to manage member revocation and attribute revocation.

VLR, which requires only to update the verifiers with revocation messages when a member is revoked, seems to be the most efficient revocation method at present. In group signature schemes, every member of a group has a token, and when he is revoked, this token is added to a list called *revocation list* (*RL*). The verifiers can check the validity of the signer using RL. In the same way, in ABGS schemes, every attribute of a member is assigned a token. Thus, when an attribute of a member is revoked, the related token is added to RL. Thus, any member with revoked attributes which are required in the policy cannot generate a valid signature.

The tokens of members are usually generated as a part of the secret signing keys in almost all the group signature schemes with VLR [16]. Thus, the adversary can attack the system if he knows the secret signing keys of the members. He can execute the verification algorithm with the tokens which he can obtain from the secret signing keys, and identify the signer. Thus, the scheme in [27] achieves weaker security notion called *selfless-anonymity* as most of the VLR group signature schemes. In selfless-anonymity, we assume that the adversary cannot get any secret signing keys. Thus, the schemes with VLR achieve the selfless-anonymity. On the other hand, VLR group signature schemes like [11,23] provided solutions to achieve stronger security than the selfless-anonymity for VLR group signature schemes. However, still, there is no Attribute-Based VLR Group Signature scheme that achieves *full-anonymity.* The full-anonymity proposed in [1] is believed to be the stronger version of anonymity. It requires to ensure the anonymity of a group signature scheme even all the member secret signing keys are exposed to an outsider.

To achieve full anonymity for ABGS with VLR, we require tokens to be independent of secret signing keys. Moreover, to prevent forging tokens, the signers should convince the verifiers that the tokens of the possessing attributes are valid, without disclosing them. As a result, we require a new zero-knowledge protocol to support such schemes.

Contribution

First, we propose a new zero-knowledge protocol which is built on the protocols given in [3, 20, 27]. Then we construct our new ABGS scheme based on the threshold-ABS scheme given in [3]. The construction of the protocol relies on the hardness of SIS and LWE lattice problems. We use decomposition, extension, masking, and permutation techniques to hide the secret data and convince the verifier that the signer has valid information. Using the Fiat-Shamir heuristic [6], we can make our new interactive protocol to non-interactive protocol.

In our scheme construction, we separate the token generation from the secret signing keys of the attributes. Since the tokens are independent of the secret signing keys, even though the secret signing keys are revealed to the adversary, he cannot attack the anonymity of the scheme. On the other hand, because of the independence of the tokens, members can fake the tokens of the attributes. To prevent such kind of forge, we require the signers to prove the nativity of the tokens while hiding them. Thus, the signers should convince the verifiers that he has relevant attributes, his attribute tokens are not being revoked, and those tokens are valid in zero-knowledge. We ensure that our new zero-knowledge protocol can satisfy those requirements.

2 Preliminary

2.1 Notation

We denote matrices by upper-case bold letters such as $\mathbf{A}$ and vectors by lower-case bold letters such as $\mathbf{v}$. Concatenation of matrices are denoted by $[\mathbf{A}|\mathbf{B}]$ and vectors by $[\mathbf{v}\|\mathbf{y}]$. For any integer $k \geq 1$, a set of integers $\{1, 2, \ldots, k\}$ is denoted by $[k]$. If S is a finite set, we present its size by $|S|$. $S(k)$ indicates its permutations of k elements and $b \hookleftarrow D$ denotes that b is sampled from a uniformly random distribution D. The standard notations of $\mathcal{O}$ and ω are used to classify the growth of functions. All algorithms are of base 2.

2.2 Discrete Gaussian Distribution

We consider a discrete Gaussian distribution for a lattice as in [3, 23].

The Gaussian function centered in a vector $\mathbf{c}$ with parameter $s > 0$ is defined as $\rho_{s,\mathbf{c}}(\mathbf{x}) = exp^{-\pi\|(\mathbf{x}-\mathbf{c})/s\|^2}$. With respect to a lattice Λ the discrete Gaussian distribution is defined as $D_{\Lambda,s,\mathbf{c}}(\mathbf{x}) = D_{s,\mathbf{c}}(\mathbf{x})/D_{s,\mathbf{c}}(\Lambda) = \rho_{s,\mathbf{c}}(\mathbf{x})/\rho_{s,\mathbf{c}}(\Lambda)$ for all $\mathbf{x} \in \Lambda$.

2.3 Lattices, Hardness of Lattices, and Lattice Related Algorithms

For n, m, and prime $q \leq 2$, let $\mathbf{B} = [\mathbf{b}_1|\cdots|\mathbf{b}_m] \in \mathbb{Z}_q^{n\times m}$ be linearly independent vectors in $\mathbb{Z}_q^n$. The n-dimensional lattice $\Lambda(\mathbf{B})$ for $\mathbf{B}$ is defined as

$$\Lambda_q^{\perp}(\mathbf{B}) = \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{B}\mathbf{x} = \mathbf{0} \mod q\},$$

$$\Lambda_q^{\mathbf{u}}(\mathbf{B}) = \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{B}\mathbf{x} = \mathbf{u} \mod q\},$$

where $\mathbf{u} \in \mathbb{Z}_q^n$.

Definition 1 (Learning With Errors (LWE)). *For integers $n, m \geq 1$, and $q \geq 2$, a vector $\boldsymbol{s} \in \mathbb{Z}_q^n$, and the Gaussian error distribution χ, the distribution $\mathrm{A}_{\boldsymbol{s},\chi}$ is obtained by sampling $\boldsymbol{a} \in \mathbb{Z}_q^n$ uniformly at random and choosing $\mathrm{e} \leftarrow \chi$, and outputting the pair $(\boldsymbol{a}, \boldsymbol{a}^T \cdot \boldsymbol{s} + \mathrm{e})$. LWE problem (decision-LWE problem) requires to distinguish LWE samples from truly random samples $\leftarrow \mathbb{Z}_q^n \times \mathbb{Z}_q$.*

For a prime power q, $b \geq \sqrt{n}\omega(\log n)$, and distribution χ, solving $LWE_{n,q,\chi}$ problem is at least as hard as solving $SIVP_\gamma$ (*Shortest Independent Vector Problem*), where $\gamma = \tilde{\mathcal{O}}(nq/b)$ [8, 24].

Definition 2 (Small Integer Solution (SIS)). *For uniformly random matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, SIS requires to find non-zero vector $\boldsymbol{x} \in \mathbb{Z}^m$, such that $\boldsymbol{A} \cdot \boldsymbol{x} = 0 \mod q$ and $\|\boldsymbol{x}\|_\infty \leq \beta$.*

Lattice Related Algorithms:

- $\mathsf{GenTrap}(n,\ m,\ q)$ takes integers $n \geq 1, q \geq 2$, and sufficiently large $m = O(n \log q)$, and outputs a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a trapdoor matrix $\mathbf{R}$. The distribution of the output $\mathbf{A}$ is negl(n)-far from the uniform distribution.
- $\mathsf{SampleD}(\mathbf{R},\ \mathbf{A},\ \mathbf{u},\ \sigma)$ takes as inputs a vector $\mathbf{u}$ in the image of $\mathbf{A}$, a trapdoor $\mathbf{R}$, and $\sigma = \omega(\sqrt{n \log q \log n})$, and outputs $\mathbf{x} \in \mathbb{Z}^m$ sampled from the distribution $D_{\mathbb{Z}^m,\sigma}$, such that $\mathbf{A} \cdot \mathbf{x} = \mathbf{u} \bmod q$.

2.4 Attribute Based Group Signature Schemes

According to the Dalia Khader's proposal [14], an ABGS scheme consists of five algorithms, namely, Setup, KeyGen, Sign, Verify, and Open. The ABGS scheme with VLR given in [27] has only former four algorithms as it employs the *implicit tracing algorithm* to track the attributes, which are used to generate a signature. The implicit tracing algorithm, which is embedded in VLR schemes, requires to execute Verify for all the user attributes until all the attributes are traced. The algorithms of a VLR-ABGS scheme are as follows.

- Setup: On input the security parameter, this algorithm sets other public parameters and defines the universal set of attributes. Then it assigns vectors for each attribute and returns all the setup parameters and set of attributes as a public parameter.
- KeyGen: On input the public parameter and the maximum number of group members, this algorithm generates a group public key and group manager's secret key. Moreover, it generates secret keys and tokens for all the attributes of all the group members. Finally, it returns the group public key, group manager's key, all the user secret signing keys, and user tokens.
- Sign: For a given policy and a message, any member who can satisfy the conditions of the policy generates a signature with his secret signing key.
- Verify: Given a message, policy and a signature, the verifier validates the signature on the message and policy and outputs 1 or 0.

2.5 Full-Anonymity

We say that an ABGS scheme is fully anonymous if no polynomial bounded adversary has a non-negligible advantage against the challenger in the bellow game.

- Init: The challenger runs Setup and KeyGen to obtain a group public key, a group manager secret key, and keys and tokens of all the attributes of all the users. Then challenger gives the group public key and all the secret signing keys of all the users to the adversary.
- Query Phase 1: The adversary requests indices of the signer and the attributes for a particular signature. He sends the signature, a message, and a policy to the challenger.
- Challenge: The challenger outputs a message, a policy, and two indices with two sets of attributes. The challenger selects one index with the related attribute set and generates a challenging signature. Then he sends the challenging signature to the adversary.
- Query Phase 2: The adversary can query of the opening of any signature as in Query Phase 1 except for the challenging signature.
- Guessing: The adversary guesses the index, which is used to generate the challenging signature. If he can guess correctly, then he wins the game.

3 Zero-Knowledge Argument of Knowledge Proof System

In this section, we propose an efficient proof of knowledge protocol which enables a prover $\mathcal{P}$ to convince the verifier $\mathcal{V}$ that he indeed a group member with a set of attributes that satisfies the given predicate Γ, and his attribute tokens are valid and are not in the revocation list RL.

We concern on statistical zero-knowledge argument systems (interactive protocols). Interactive protocols have two properties called *soundness property* and *zero-knowledge property.* While the soundness property only holds for *computationally bounded* cheating provers, the zero-knowledge property holds against *any* cheating verifiers [20].

We are engaging with *string commitment scheme*, which uses a string as the committed value and which satisfies the above requirements. Kawachi et al. [12] presented a more straightforward construction from lattices for string commitment scheme **COM**. Later, using the Kawachi's string commitment scheme, Ling et al. [19] proposed a Stern type zero-knowledge proof of knowledge for lattices. The security of their protocol is based on the hardness of the underlying ISIS (Inhomogeneous SIS) problem. In other words, to break their protocol, an attacker needs to solve the underlying ISIS problem. Ling et al. [19] achieved security by using a technique called *Decomposition-Extension.*

3.1 Techniques

We define some techniques that were used in the existing protocols [16, 20, 27], and which we use in the construction of our protocol.

- **Decomposition-Extension Technique**
  Let $k = \lfloor \log \beta \rfloor$ and the sequence of integers $\beta_1, \ldots, \beta_k$ be as follows.
  $\beta_1 = \lceil \beta/2 \rceil; \beta_2 = \lceil (\beta - \beta_1)/2 \rceil; \beta_3 = \lceil (\beta - \beta_1 - \beta_2)/2 \rceil; \ldots ; \beta_k = 1$.
  Ling et al. [19] observed that an integer $z \in [0, \beta]$, if and only if there exists $z_1, \ldots, z_k \in \{0, 1\}$ such that $z = \sum_{j=1}^{k} \beta_j z_j$.
  The above observation allows the prover to efficiently decompose $\mathbf{z} \in [-\beta; \beta]^m$ into $\tilde{\mathbf{z}}_1, \ldots, \tilde{\mathbf{z}}_k \in \{-1, 0, 1\}^m$ such that $\sum_{j=1}^{k} \beta_j \tilde{\mathbf{z}}_j = \mathbf{z}$. To extend a vector $\tilde{\mathbf{z}}$ to $\mathbf{z} \in \mathsf{B}_{3m}$, where B_{3m} is a set of vectors in $\{-1, 0, 1\}^{3m}$ having exactly m coordinates equal to -1, m coordinates equal to 0, and m coordinates equal to 1, we select a random vector $\hat{\mathbf{z}} \in \{-1, 0, 1\}^{2m}$, and output $\mathbf{z} = (\tilde{\mathbf{z}} \| \hat{\mathbf{z}})$. Here $\hat{\mathbf{z}} \in \{-1, 0, 1\}^{2m}$ has $(m - \lambda_{-1})$ coordinates equal to -1, $(m - \lambda_0)$ coordinates equal to 0, and $(m - \lambda_1)$ coordinates equal to 1.
- **Matrix-Extension Technique**
  For a given matrix $\bar{\mathbf{A}}$ the extended matrix $\bar{\mathbf{A}}^*$ is obtained by appending $2m$ $zero-columns$ to the matrix $\bar{\mathbf{A}}$. For instance, if the given matrix $\bar{\mathbf{A}} = [\mathbf{A} | \mathbf{A}_0 | \mathbf{A}_1 | \ldots | \mathbf{A}_\ell] \in \mathbb{Z}_q^{n \times (2+\ell)m}$, then the extended matrix $\bar{\mathbf{A}}^* \in \mathbb{Z}_q^{n \times (2+2\ell)3m}$ is obtained as

  $$\bar{\mathbf{A}}^* = [\mathbf{A} | 0^{n \times 2m} | \mathbf{A}_0 | 0^{n \times 2m} | \ldots | \mathbf{A}_\ell | 0^{n \times 2m} | 0^{n \times 3m\ell}].$$

  Using the above techniques, in Stern protocol, the prover $\mathcal{P}$ can convince the verifier $\mathcal{V}$ that $\mathbf{z} \in [-\beta, \beta]^m$ and $\mathbf{A}\mathbf{z} = \mathbf{A}^* \sum_{j=1}^{k} \beta_j \mathbf{z}_j = \mathbf{u} \mod q$ by demonstrating below two statements.
  1. For each j, a random permutation of $\mathbf{z}_j$ belongs to B_{3m}. Thus, $\mathbf{z}_j \in \mathsf{B}_{3m}$ and $\tilde{\mathbf{z}}_j \in \{-1, 0, 1\}^m$. This will convince that $\mathbf{z} \in [-\beta, \beta]^m$.
  2. $\mathbf{A}^* \sum_{j=1}^{k} \beta_j (\mathbf{z}_j + \mathbf{r}_j) - \mathbf{u} = \mathbf{A}^* \sum_{j=1}^{k} \beta_j \mathbf{r}_j \mod q$, where $\mathbf{A}^*$ is the extended matrix of $\mathbf{A}$ and $\mathbf{r}_1, \ldots, \mathbf{r}_k \in \mathbb{Z}_q^{3m}$ are uniformly "masking" vectors for $\mathbf{z}_j$. This convinces that $\mathbf{A}\mathbf{z} = \mathbf{A}^* \sum_{j=1}^{k} \beta_j \mathbf{z}_j = \mathbf{u} \mod q$.

- For permutations $\pi, \psi \in \mathsf{S}_{3m}$; $\tau \in \mathsf{S}_{2\ell}$, $\xi \in \mathsf{S}_p$, and for a vector $\mathbf{z} = (\mathbf{z}_{-1} \| \mathbf{z}_0 \| \mathbf{z}_1 \| \ldots \| \mathbf{z}_{2\ell}) \in \mathbb{Z}_q^{(2+2\ell)3m}$ we define,
$$F_{\pi,\psi,\tau,\xi}(\mathbf{z}) = (\pi(\mathbf{z}_{\xi(-1)}) \| \psi(\mathbf{z}_{\xi(0)}) \| \psi(\mathbf{z}_{\xi,\tau(1)}) \| \ldots \| \psi(\mathbf{z}_{\xi,\tau(2\ell)})).$$
$F_{\pi,\psi,\tau,\xi}(\mathbf{z})$ *rearranges* the order of $2{+}2\ell$ blocks $\mathbf{z}_{-1}, \mathbf{z}_0, \ldots, \mathbf{z}_{2\ell}$ according to ξ and the order of 2ℓ blocks $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{2\ell}$ according to τ. Then it *permutes* block $\mathbf{z}_{-1}$ according to π and the other $(1{+}2\ell)$ blocks according to ψ.
- For a given $\bar{\mathbf{z}} = (\mathbf{x} \| \mathbf{y} \| d_1\mathbf{y} \| \ldots \| d_\ell \mathbf{y}) \in \mathbb{Z}^{(2+\ell)m}$, we say, $d \in \{0,1\}^\ell$, if $d^* = (d_1, \ldots, d_\ell, d_{\ell+1}, \ldots, d_{2\ell}) \in B_{2\ell}$ and the random permutation of d^* is in the set of $B_{2\ell}$, where d^* is the extension of d and $B_{2\ell}$ is the set of vectors in $\{0,1\}^{2\ell}$ having Hamming weight ℓ.
- We say, $\mathbf{z} \in \mathsf{VALID}(d^*)$ if $\mathbf{z} \in \{-1,0,1\}^{(2+2\ell)3m}$ and there exits $\mathbf{x}, \mathbf{y} \in \mathsf{B}_{3m}$, such that $\mathbf{z} = (\mathbf{x} \| \mathbf{y} \| d_1\mathbf{y} \| d_2\mathbf{y} \| \ldots \| d_{2\ell}\mathbf{y})$.

Based on the above discussion, we build our ZK-proof system.

3.2 Underlying Interactive Protocol

For an attribute i that a user has, we assign a vector $\mathbf{z}_i$ sampled from $D_{\mathbb{Z}^{2m},\sigma}$, which satisfies $\|\mathbf{z}_i\|_\infty \leq \beta$. For an attribute i that the user does not have, we assign a vector $\mathbf{z}_i$ sampled from $D_{\mathbb{Z}^{2m},\sigma}$, which does not satisfy $\|\mathbf{z}_i\|_\infty \leq \beta$.

Suppose a user with index d possesses valid credentials for a set of attributes $S_d = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_a\}$ and the given predicate is $\mathit{\Gamma} = \{t, S \subseteq \mathrm{Att}, t \in \mathbb{N} \wedge (S = \mathbf{u}_1, \ldots, \mathbf{u}_p)\}$, where Att is the universal set of attributes $\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_u\}$ and $\mathit{\Gamma}$ requires the signer to satisfy at least t attributes out of S. Let $S_m = S \cap S_d$ and $S_r = S \setminus S_m$, where $|S_m| = t$ and $|S| = p - t$.

- The public parameters are: a matrix $(\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell) \in \mathbb{Z}_q^{n \times (2+\ell)m}$, a set of vectors $\{\mathbf{u}_i\}_{i=1}^p$, a threshold predicate $\mathit{\Gamma} = (t, S)$, matrices $\{\mathbf{B}_i \in \mathbb{Z}_q^{m \times n}\}_{i=1}^p$, and vectors $\{\mathbf{b}_i \in \mathbb{Z}_q^m\}_{i=1}^p$, where $t \leq |S| = p$.
- The prover's witnesses are: the index $d \in \{0,1\}^\ell$, t vectors $\mathbf{z}_i = (\mathbf{x} \| \mathbf{y} \| d_1\mathbf{y} \| \ldots \| d_\ell\mathbf{y})$ $\mathbf{u}_i \in S_m$, where $\|\mathbf{z}_i\|_\infty \leq \beta$, $p - t$ vectors $\mathbf{z}_i = (\mathbf{x} \| \mathbf{y} \| d_1\mathbf{y} \| \ldots \| d_\ell\mathbf{y})$ for $\mathbf{u}_i \in S_r$, p vectors $\mathbf{t}_i \in \mathbb{Z}^m$, and p vectors $\mathbf{e}_i \in \mathbb{Z}^m$.
- The prover's goal is to convince the verifier in zero knowledge that:
  - For $i \in [t]$, $\mathbf{A}_d \mathbf{z}_i = \mathbf{u}_i \mod q$ and $\|\mathbf{z}_i\|_\infty \leq \beta$, where $\mathbf{A}_d = [\mathbf{A} | \mathbf{A}_0 + \sum_{i=1}^{\ell} d_i \mathbf{A}_i]$.
  - For $i \in [p-t]$, $\mathbf{A}_d \mathbf{z}_i = \mathbf{u}_i \mod q$ and $\|\mathbf{z}_i\|_\infty \nleq \beta$.
  - For $i \in [p]$, $\|\mathbf{e}_i\| \leq \beta$ and $\mathbf{B}_i \cdot (\mathbf{A} \cdot \mathbf{t}_i) + \mathbf{e}_i = \mathbf{b}_i \mod q$.
  - For $i \in [p]$, $(\mathbf{A} \cdot \mathbf{t}_i) + (\mathbf{A}'_d \cdot \mathbf{z}_i) = \mathbf{u}_i \mod q$, where $\mathbf{A}'_d = [0 \in \mathbb{Z}_q^{n \times m} \mid 0 \in \mathbb{Z}_q^{n \times m} + \sum_{i=1}^{\ell} d_i \cdot \mathbf{A}_i]$.

Both the prover $\mathcal{P}$ and the verifier $\mathcal{V}$ compute the following matrices.

- $\bar{\mathbf{A}}^* = [\mathbf{A} | 0 \in \mathbb{Z}^{n \times 2m} | \mathbf{A}_0 | 0 \in \mathbb{Z}^{n \times 2m} | \ldots | \mathbf{A}_\ell | 0 \in \mathbb{Z}^{n \times 2m} | 0 \in \mathbb{Z}^{2 \times 3m\ell}] \in \mathbb{Z}_q^{n \times (2+2\ell)3m}$.

- $\{(\mathbf{B}_i^* = \mathbf{B}_i \cdot \mathbf{A}) \in \mathbb{Z}_q^{m\times m}\}_{i=1}^{p}$.
- $\{\mathbf{I}_i^* \in \{0,1\}^{m\times 3m}\}_{i=1}^{p}$. Each matrix is obtained by appending $2m$ $zero-columns$ to the identity matrix of order m.
- $\bar{\mathbf{A}}'^* = [0 \in \mathbb{Z}^{n\times 3m}|0 \in \mathbb{Z}^{n\times 3m}|\mathbf{A}_1|0 \in \mathbb{Z}^{n\times 2m}|\ldots|\mathbf{A}_\ell|0 \in \mathbb{Z}^{n\times 2m}|0 \in \mathbb{Z}^{2\times 3m\ell}] \in \mathbb{Z}_q^{n\times(2+2\ell)3m}$.

Then,

- For S_m, the prover $\mathcal{P}$ applies the Decomposition-Extension technique on $\mathbf{z}_i$, and generates masking terms $\{\mathbf{r}_{z(i)}^j\}$, where $i \in [t]$ and $j \in [k]$, such that the verifier can check $\bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \cdot (\mathbf{z}_i^j + \mathbf{r}_{z(i)}^j)) - \mathbf{u}_i = \bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{z(i)}^j) \mod q$, where $\mathbf{z}_i^j \in \mathsf{VALID}(d^*)$.
- For S_r, $\mathcal{P}$ decomposes, extends $\mathbf{z}_i$, and generates masking terms $\{\mathbf{r}_{z(i)}^j\}$, where $i \in [p-t]$ and $j \in [k]$, such that $\bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \cdot (\mathbf{z}_i^j + \mathbf{r}_{z(i)}^j)) - \mathbf{u}_i = \bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{z(i)}^j) \mod q$.
- For S, $\mathcal{P}$ decomposes, extends both $\mathbf{t}_i$ and $\mathbf{e}_i$, and generates masking terms $\{\mathbf{r}_{t(i)}^j\}$, where $i \in [p]$ and $j \in [k]$, and $\{\mathbf{r}_{e(i)}^j\}$, where $i \in [p]$ and $j \in [k]$ respectively, such that $(\mathbf{B}_i^* \cdot (\sum_{j=1}^k \beta_j \cdot (\mathbf{t}_i^j + \mathbf{r}_{t(i)}^j)) + \mathbf{I}_i^* \cdot (\sum_{j=1}^k \beta_j \cdot (\mathbf{e}_i^j + \mathbf{r}_{e(i)}^j))) - \mathbf{b}_i = \mathbf{B}_i^* \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{t(i)}^j) + \mathbf{I}_i^* \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{e(i)}^j) \mod q$.
- Similarly, $(\mathbf{A} \cdot (\sum_{j=1}^k \beta_j \cdot (\mathbf{t}_i^j + \mathbf{r}_{t(i)}^j)) + \bar{\mathbf{A}}'^* \cdot (\sum_{j=1}^k \beta_j \cdot (\mathbf{z}_i^j + \mathbf{r}_{z(i)}^j))) - \mathbf{u}_i = \mathbf{A} \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{t(i)}^j) + \bar{\mathbf{A}}'^* \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{z(i)}^j) \mod q$.

Description of the Protocol

Commitments:

- Randomly sample masking terms $\{\mathbf{r}_{z(i)}^j \hookleftarrow \mathbb{Z}_q^{(2+2\ell)3m}, \mathbf{r}_{t(i)}^j \hookleftarrow \mathbb{Z}_q^m, \mathbf{r}_{e(i)}^j \hookleftarrow \mathbb{Z}_q^{3m}\}^{p\cdot k}$ for $i \in [p], j \in [k]$ and $\mathbf{r}_{d^*} \hookleftarrow \mathbb{Z}_q^{2\ell}$.
- Sample permutations $\{\pi_j, \psi_j \hookleftarrow S_{3m}, \phi_j, \hookleftarrow S_m, \varphi_j, \hookleftarrow S_{3m}\}_{j=1}^{p\cdot k}$, $\tau \hookleftarrow S_{2\ell}$, and $\xi \hookleftarrow S_p$.

The prover $\mathcal{P}$ generates commitments CMT $= (\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3)$, and sends to the verifier $\mathcal{V}$.

- $\mathbf{c}_1 = \mathbf{COM}(\tau, \xi, \{\pi_j, \psi_j, \phi_j, \varphi_j\}_{j=1}^{p\cdot k}, \{\bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \mathbf{r}_{z(i)}^j)\}_{i\in[p]},$
  $\{\mathbf{B}_i^* \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{t(i)}^j) + \mathbf{I}_i^*(\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{e(i)}^j)\}_{i\in[p]},$
  $\{\mathbf{A} \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}_{t(i)}^j) + \bar{\mathbf{A}}'^*(\sum_{j=1}^k \mathbf{r}_z^j)\}_{i\in[p]})$.
- $\mathbf{c}_2 = \mathbf{COM}(\tau(\mathbf{r}_{d^*}), \{\{F_{\pi_i^j, \psi_i^j, \tau, \xi}(\mathbf{r}_{z(i)}^j)\}_{j=1}^k\}_{i\in[p]}, \{\{\phi_i^j(\mathbf{r}_{t(i)}^j)\}_{j=1}^k\}_{i\in[p]},$
  $\{\{\varphi_i^j(\mathbf{r}_{e(i)}^j)\}_{j=1}^k\}_{i\in[p]})$.
- $\mathbf{c}_3 = \mathbf{COM}(\tau(d^* + \mathbf{r}_{d^*}), \{\{F_{\pi_i^j, \psi_i^j, \tau, \xi}(\mathbf{z}_i^j + \mathbf{r}_{z(i)}^j)\}_{j=1}^k\}_{i\in[p]},$
  $\{\{\phi_i^j(\mathbf{t}_i^j + \mathbf{r}_{t(i)}^j)\}_{j=1}^k\}_{i\in[p]}, \{\{\varphi_i^j(\mathbf{e}_i^j + \mathbf{r}_{e(i)}^j)\}_{j=1}^k\}_{i\in[p]})$.

Challenge: The verifier $\mathcal{V}$ randomly chooses a challenge $CH \hookleftarrow \{1,2,3\}$, and sends it to $\mathcal{P}$.

Response: Depending on the challenge CH, the prover $\mathcal{P}$ responses as below.

- $CH = 1$: Let $\mathbf{v}_{d^*} = \tau(d^*)$ and $\mathbf{w}_{d^*} = \tau(\mathbf{r}_{d^*})$.
  For $i \in [p]$ let
  $\{\mathbf{v}^j_{z(i)} = F_{\pi_i^j,\psi_i^j,\tau,\xi}(\mathbf{z}_i^j)\}_{j=1}^k, \{\mathbf{w}^j_{z(i)} = F_{\pi_i^j,\psi_i^j,\tau,\xi}(\mathbf{r}^j_{z(i)})\}_{j=1}^k,$
  $\{\mathbf{v}^j_{t(i)} = \phi_i^j(\mathbf{t}_i^j)\}_{j=1}^k, \{\mathbf{w}^j_{t(i)} = \phi_i^j(\mathbf{r}^j_{t(i)})\}_{j=1}^k,$
  $\{\mathbf{v}^j_{e(i)} = \varphi_i^j(\mathbf{e}_i^j)\}_{j=1}^k, \{\mathbf{w}^j_{e(i)} = \varphi_i^j(\mathbf{r}^j_{e(i)})\}_{j=1}^k.$
  Output $RSP_1 = (\mathbf{v}_{d^*}, \mathbf{w}_{d^*}, \{\{\mathbf{v}^j_{z(i)}, \mathbf{w}^j_{z(i)}, \mathbf{v}^j_{t(i)}, \mathbf{w}^j_{t(i)}, \mathbf{v}^j_{e(i)}, \mathbf{w}^j_{e(i)}\}_{j=1}^k\}_{i\in[p]})$.
- $CH = 2$: Let $\mathbf{y}_{d^*} = d^* + \mathbf{r}_{d^*}$.
  For $i \in [p]$ let $\{\{\mathbf{y}^j_{z(i)} = \mathbf{z}_i^j + \mathbf{r}^j_{z(i)}\}_{j=1}^k$, $\{\mathbf{y}^j_{t(i)} = \mathbf{t}_i^j + \mathbf{r}^j_{t(i)}\}_{j=1}^k$,
  $\{\mathbf{y}^j_{e(i)} = \mathbf{e}_i^j + \mathbf{r}^j_{e(i)}\}_{j=1}^k\}$.
  Output $RSP_2 = (\tau, \xi, \{\pi_j, \psi_j, \phi_j, \varphi_j\}_{j=1}^{p\cdot k}, \mathbf{y}_{d^*}, \{\{\mathbf{y}^j_{z(i)}, \mathbf{y}^j_{t(i)}, \mathbf{y}^j_{e(i)}\}_{j=1}^k\}_{i\in[p]})$.
- $CH = 3$:
  Output $RSP_3 : (\tau, \xi, \{\pi_j, \psi_j, \phi_j, \varphi_j\}_{j=1}^{p\cdot k}, \mathbf{r}_{d^*}, \{\{\mathbf{r}^j_{z(i)}, \mathbf{r}^j_{t(i)}, \mathbf{r}^j_{e(i)}\}_{j=1}^k\}_{i\in[p]})$.

Verification: The verifier $\mathcal{V}$ checks the received response RSP as follows.

- $CH = 1$: Check that $\mathbf{v}_{d^*} \in B_{2\ell}$, $\mathbf{v}^j_{z(i)}$ is valid with respect to $\mathbf{v}_{d^*}$ (that is $\mathbf{v}^j_{z(i)} \in \mathsf{VALID}(\mathbf{v}_{d^*})$) for at least t set of vectors and all $j \in [k]$, $\mathbf{v}^j_{t(i)} \in B_m$, and $\mathbf{v}^j_{e(i)} \in B_{3m}$. Then check that,
  - $\mathbf{c}_2 = \mathbf{COM}(\mathbf{w}_{d^*}, \{\{\mathbf{w}^j_{z(i)}, \mathbf{w}^j_{t(i)}, \mathbf{w}^j_{e(i)}\}_{j=1}^k\}_{i\in[p]})$,
  - $\mathbf{c}_3 = \mathbf{COM}((\mathbf{v}_{d^*} + \mathbf{w}_{d^*}), \{\{(\mathbf{v}^j_{z(i)} + \mathbf{w}^j_{z(i)}), (\mathbf{v}^j_{t(i)} + \mathbf{w}^j_{t(i)}),$
    $(\mathbf{v}^j_{e(i)} + \mathbf{w}^j_{e(i)})\}_{j=1}^k\}_{i\in[p]})$.
- $CH = 2$: Check that
  - $\mathbf{c}_1 = \mathbf{COM}(\tau, \xi, \{\pi_j, \psi_j, \phi_j, \varphi_j\}_{j=1}^{p\cdot k}, \{\bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \mathbf{y}^j_{z(i)}) - \mathbf{u}_i\}_{i\in[p]},$
    $\{\mathbf{B}_i^*(\sum_{j=1}^k \beta_j \cdot \mathbf{y}^j_{t(i)}) + \mathbf{I}_i^*(\sum_{j=1}^k \beta_j \cdot \mathbf{y}^j_{e(i)}) - \mathbf{b}_i\}_{i\in[p]}$
    $\{\mathbf{A} \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{y}^j_{t(i)}) + \bar{\mathbf{A}}'^*(\sum_{j=1}^k \mathbf{y}^j_{z(i)}) - \mathbf{u}_i\}_{i\in[p]})$,
  - $\mathbf{c}_3 = \mathbf{COM}(\tau(\mathbf{y}_{d^*}), \{\{F_{\pi_i^j,\psi_i^j,\tau,\xi}(\mathbf{y}^j_{z(i)})\}_{j=1}^k\}_{i\in[p]},$
    $\{\{\phi_i^j(\mathbf{y}^j_{t(i)}\}_{j=1}^k\}_{i\in[p]}, \{\{\varphi_i^j(\mathbf{y}^j_{e(i)}\}_{j=1}^k\}_{i\in[p]})$.
- $CH = 3$: Check that
  - $\mathbf{c}_1 = \mathbf{COM}(\tau, \xi, \{\pi_j, \psi_j, \phi_j, \varphi_j\}_{j=1}^{p\cdot k}, \{\bar{\mathbf{A}}^* \cdot (\sum_{j=1}^k \beta_j \mathbf{r}^j_{z(i)})\}_{i\in[p]},$
    $\{\mathbf{B}_i^*(\sum_{j=1}^k \beta_j \cdot \mathbf{r}^j_{t(i)}) + \mathbf{I}_i^*(\sum_{j=1}^k \beta_j \cdot \mathbf{r}^j_{e(i)})\}_{i\in[p]},$
    $\{\mathbf{A} \cdot (\sum_{j=1}^k \beta_j \cdot \mathbf{r}^j_{t(i)}) + \bar{\mathbf{A}}'^*(\sum_{j=1}^k \mathbf{r}^j_{z(i)})\}_{i\in[p]})$,
  - $\mathbf{c}_2 = \mathbf{COM}(\tau(\mathbf{r}_{d^*}), \{\{F_{\pi_i^j,\psi_i^j,\tau,\xi}(\mathbf{r}^j_{z(i)})\}_{j=1}^k\}_{i\in[p]}$, $\{\{\phi_i^j(\mathbf{r}^j_{t(i)})\}_{j=1}^k\}_{i\in[p]},$
    $\{\{\varphi_i^j(\mathbf{r}^j_{e(i)})\}_{j=1}^k\}_{i\in[p]}$.

$\mathcal{V}$ outputs 1 if and only if all the conditions hold, otherwise he outputs 0.

3.3 Analysis of the Protocol

Theorem 1. *Let COM be a statistically hiding and computationally binding string commitment scheme. Then our protocol in Section 3.2 is a zero-knowledge argument of knowledge for the relation* $\mathrm{R} = (n, \ell, m, t, p, k, \beta)$ *with perfect completeness, soundness error 2/3, and communication cost* $(\mathcal{O}(p\ell m \log \beta) \log q$.

Completeness and Communication Cost. If the prover $\mathcal{P}$ is honest and follows the protocol, then the verifier $\mathcal{V}$ always outputs 1. Based on the previous discussion, the proposed protocol has perfect completeness. Moreover, according to [12], the commitment CMT has $3n \log q$ bits. The verifier $\mathcal{V}$ sends two-bit challenge $CH \in \{1, 2, 3\}$. The response RSP of $\mathcal{P}$ is a subset of the set of masking terms and permutations which sums overall communication cost of upper bound $\mathcal{O}(p\ell m \log \beta) \log q$.

We employ standard simulation and extraction techniques for Stern-like protocol [12,19,26] to prove the proposed protocol is a ZKAoK. The detailed proof is given in the full version of this paper.

4 Proposed Attribute-Based VLR Group Signature Scheme

Let λ be the security parameter, and $N = 2^\ell = poly(\lambda)$ be the maximum number of members in a group. Let integer $n = poly(\lambda)$, the modulus $q = \mathcal{O}(\ell n^2)$, and the dimension $m = \lceil 2n \log q \rceil$. Gaussian parameter $\sigma = \omega(\log m)$. The infinity norm bound for signature is $\beta = \tilde{\mathcal{O}}(\sqrt{\ell n})$.

- $\mathsf{Setup}(1^\lambda)$: On input the security parameter λ, set the parameters *para* as above, and proceed as below.
  1. Define the universal set of attributes $Att = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_u\}$, where $\mathbf{u}_i \in \mathbb{Z}_q^n$ is uniform random and $|Att| = u$. Each attribute att_i is associated to a uniform random vector $\mathbf{u}_i$ via a list $attLst = \{(\mathrm{att}_i, \mathbf{u}_i)\}_{i \in \{1,2,\ldots,u\}}$.
  2. Select a hash function $\mathcal{H} : \{0,1\}^* \rightarrow \{1,2,3\}^t$, to be modeled as a random oracle, where $t = \omega(\log n)$.
  3. Output the public parameters $PP = (para, Att, attLst, \mathcal{H})$.
- $\mathsf{KeyGen}(PP, N)$: The randomized algorithm KeyGen takes the public parameters PP and $N = 2^\ell$ as the inputs and works as follows.
  1. Generate the verification key $\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n \times m}$ and a trapdoor $\mathbf{T_A} \in \mathbb{Z}_q^{m \times m}$ for the modified Boyen's signature scheme as in [22].
  2. For a member with an index $d \in \{0, 1, \ldots, N-1\}$ and a set of attributes $S_d = \{\mathbf{u}_{a_1}, \mathbf{u}_{a_2}, \ldots, \mathbf{u}_{a_s}\} \subseteq \mathrm{Att}$ $(|S_d| = s)$, execute the following steps to generate keys and tokens for him.
     (a) Let $d[1] \ldots d[\ell] \in \{0,1\}^\ell$ be the binary representation of d.
     (b) Compute $\mathbf{A}_d = [\mathbf{A} \mid \mathbf{A}_0 + \sum_{i=1}^{\ell} d[i] \cdot \mathbf{A}_i] \in \mathbb{Z}^{n \times 2m}$.
     (c) For all $j \in \{1, 2, \ldots, s\}$ sample $\mathbf{z}_{d,a_j} \hookleftarrow D_{\mathbb{Z}^{2m},\sigma}$ as the secret key for an attribute $\mathbf{u}_{a_j}$ such that $\mathbf{A}_d \cdot \mathbf{z}_{d,a_j} = \mathbf{u}_{a_j}$ and $\|\mathbf{z}_{d,a_j}\| \leq \beta$.

   (d) For the other attributes $u - s$ again sample fake credentials $\mathbf{f}_{d,f_j} \hookleftarrow D_{\mathbb{Z}^{2m},\sigma}$, such that $\mathbf{A}_d \cdot \mathbf{f}_{d,f_j} = \mathbf{u}_j$ and $\|\mathbf{f}_{d,f_j}\| \nleq \beta$.
   (e) Hereafter we represent all the secret keys (fake or real) for attributes by $\mathbf{z}_{d,a_j}$.
   (f) Get $\mathbf{A}'_d = [0 \in \mathbb{Z}_q^{n\times m} \mid 0 \in \mathbb{Z}_q^{n\times m} + \sum_{i=1}^{\ell} d[i] \cdot \mathbf{A}_i]$ by replacing $\mathbf{A}$ and $\mathbf{A}_0$ with zero matrices in the step (b).
   (g) Compute $\mathbf{v}_{d_j} = \mathbf{A}'_d \cdot \mathbf{z}_{d_j} \in \mathbb{Z}^n$ for all the attributes.
   (h) Run $\mathsf{SampleD}(\mathbf{T_A}, \mathbf{A}, \mathbf{u}_j - \mathbf{v}_{d_j}, \sigma)$ to obtain $\mathbf{t}_{d_j}$ for all the attributes.
   (i) Let the secret signing key of d be $\mathbf{gsk}[d] = \{\mathbf{z}_{d_j}, \mathbf{u}_j\}_{j\in[u]}$ and the revocation token be $\mathbf{grt}[d] = \{\mathbf{u}^t_{d_j} = \mathbf{A} \cdot \mathbf{t}_{d_j}\}_{j\in[u]}$.

3. Output the group public key $\mathbf{gpk} = (\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_u)$, the group manager's secret key $\mathbf{gmsk} = \mathbf{T_A}$, the members' secret signing keys **gsk**= (**gsk**[0], **gsk**[1], . . . , **gsk**[N-1]), and members' revocation tokens **grt** = (**grt**[0], **grt**[1], . . . , **grt**[N-1]).

– $\mathsf{Sign}(PP, \mathit{\Gamma}, \mathbf{gpk}, \mathbf{gsk}[d], \mathbf{grt}[d], S_d, M)$: On input the group public key **gpk**, and a message M, the user d in a possession of a secret signing key $\mathbf{gsk}[d]$, a revocation token $\mathbf{grt}[d]$, and a set of attributes $S_d \subseteq$ Att, generates a signature for a given threshold predicate $\mathit{\Gamma} = (t, S = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_p\} \subseteq$ Att), where $1 \leq t \leq |S| = p$, as below. Here, $\mathit{\Gamma} = (t, S)$ implies that the condition (policy) $\mathit{\Gamma}$ requires the signer to posses at least t attributes out of the given set of attributes S, where the size of S is p.

1. Let $S_m \subseteq (S \cap S_d) \subseteq Att$ be the matching attributes that the user d possesses, where $|S_m| = t$.
2. For the attributes $S \setminus S_m$ the user d has fake credentials.
3. For all the attributes $i \in p$,
   (a) Sample $\rho_i \xleftarrow{\$} \{0,1\}^n$, let $\mathbf{B}_i = \mathcal{G}(\bar{\mathbf{A}}, \mathbf{u}_i, M, \rho_i) \in \mathbb{Z}_q^{n\times m}$ ($\mathcal{G} : \{1,2,3\}^* \to \mathbb{Z}_q^{n\times m}$), where $\bar{\mathbf{A}} = [\mathbf{A}|\mathbf{A}_0|\ldots|\mathbf{A}_\ell]$.
   (b) Compute $\mathbf{b}_i = \mathbf{B}_i \cdot (\mathbf{A} \cdot \mathbf{t}_{d_i}) + \mathbf{e}_i \mod q$ ($\|\mathbf{e}_i\|_\infty \leq \beta$ with overwhelming probability).
4. Generate a non-interactive zero-knowledge argument of knowledge $\mathit{\Pi}$ to prove that the prover d is indeed a valid group member possessing at least t non-revoked attributes among $S \subseteq$ Att. This is done by repeating the protocol given in Section 3, $\bar{t} = \omega(\log n)$ times with public inputs $(\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \{\mathbf{u}_i\}_{i\in[p]}, \{\mathbf{B}_i\}_{i\in[p]}, \{\mathbf{b}_i\}_{i\in[p]})$ and witness $(d, \{\mathbf{z}_i\}_{i\in[p]}, \{\mathbf{t}_i\}_{i\in[p]}, \{\mathbf{e}_i\}_{i\in[p]})$. Then make it non-interactive via the Fiat-Shamir heuristic as a triple $\mathit{\Pi} = (\{\mathrm{CMT}^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}}, \mathrm{CH}, \{\mathrm{RSP}^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}})$, where $\mathrm{CH} = (\{Ch^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}}) = \mathcal{H}(M, \mathbf{A}, \{\mathbf{A}_i\}_{i=0}^{\ell}, \{\mathbf{u}_i\}_{i=1}^{p}, \{\mathbf{B}_i\}_{i=1}^{p}, \{\mathbf{b}_i\}_{i=1}^{p}, \{\mathrm{CMT}^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}})$.
5. Output a signature $\mathit{\Sigma} = (M, \{\rho_i\}_{i=1}^{p}, \{\mathbf{b}_i\}_{i=1}^{p}, \mathit{\Pi})$.

– $\mathsf{Verify}(PP, \mathit{\Gamma}, \mathbf{gpk}, RL, M, \mathit{\Sigma})$: This deterministic algorithm takes as inputs the group public key $\mathbf{gpk} = (\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_u)$, a threshold predicate $\mathit{\Gamma} = (t, S = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_p\} \subseteq$ Att), a signature $\mathit{\Sigma}$ on a message M, and a list of revocation tokens $RL = \{\mathbf{u}^t_i = (\mathbf{u}^t_{i_1}, \mathbf{u}^t_{i_2}, \ldots, \mathbf{u}^t_{i_a})\}_{i\leq N} \subseteq \mathbf{grt}$, where $a \leq u$, and verifies the signature as below.

1. Pares the signature Σ as $(M, \{\rho_i\}_{i=1}^{p}, \{\mathbf{b}_i\}_{i=1}^{p}, \Pi)$.
2. Get $\{\mathbf{B}_i = \mathcal{G}(\bar{\mathbf{A}}, \mathbf{u}_i, M, \rho_i) \in \mathbb{Z}_q^{n\times m}\}_{i\in[p]}$.
3. Pares Π as $(\{\mathrm{CMT}^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}}, \{Ch^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}}, \{\mathrm{RSP}^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}})$.
4. Return 0, if $(Ch_1, \ldots Ch_{\bar{t}}) \neq$ $\mathcal{H}(M, \mathbf{A}, \{\mathbf{A}_i\}_{i=0}^{\ell}, \{\mathbf{u}_i\}_{i=1}^{p}, \{\mathbf{B}_i\}_{i=1}^{p}, \{\mathbf{b}_i\}_{i=1}^{p}, \{\mathrm{CMT}^{(\bar{k})}\}_{\bar{k}=1}^{\bar{t}})$.
5. For $i = 0$ to $\bar{t}$, run the verification steps of the protocol given in Section 3 with the public inputs $(\mathbf{A}, \mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \{\mathbf{u}_i\}_{i\in[p]}, \{\mathbf{B}_i\}_{i\in[p]}, \{\mathbf{b}_i\}_{i\in[p]})$ to check the validity of $\mathrm{RSP}^{(\bar{k})}$ with respect to $\mathrm{CMT}^{(\bar{k})}$ and $Ch^{(\bar{k})}$. If any of the conditions does not hold, then return 0.
6. For each $\mathbf{u}_{i_x}^t$ in the given revocation list RL, where $x \leq u$ and $i \leq N$ compute $\mathbf{e}_i^{'} = \mathbf{b}_i - \mathbf{B}_i \cdot \mathbf{u}_{i_x}^t \mod q$ to check whether there exists an index i such that $\|\mathbf{e}_i^{'}\|_\infty \leq \beta$. If so return 0.
7. Return 1.

– $\mathsf{Revoke}(PP, \mathbf{gpk}, \mathbf{gmsk}, RL, d, S_r)$: On input $\mathbf{gpk}$, the revocation list RL, the id d of the effecting member, and his revoking attribute set $S_r = \{\mathbf{u}_{d_1}^t = \mathbf{A}\cdot\mathbf{t}_{d_1}, \mathbf{u}_{d_2}^t = \mathbf{A}\cdot\mathbf{t}_{d_2}, \ldots, \mathbf{u}_{d_r}^t = \mathbf{A}\cdot\mathbf{t}_{d_r}\}$, where $r \leq u$, the group manager with $\mathbf{gmsk}$, do the following steps.
  1. If $\mathbf{u}_d^t \in RL$, then $\mathbf{u}_d^t = \mathbf{u}_d^t \cup S_r$, else $RL = RL \cup \mathbf{u}_d^t = S_r$.
  2. Return RL.

5 Security Analysis of the Proposed Scheme

This paper provides a new ABGS scheme with VLR from lattices to achieve full-anonymity. The security of the scheme is proven in the random-oracle model under the hardness assumption of SIVP problem.

Theorem 2. *The proposed ABGS-VLR is correct with overwhelming probability. If the underlying non-interactive zero-knowledge (NIZK) protocol is simulation sound and zero-knowledge, then the proposed scheme is fully anonymous. Moreover, under the hardness of the $\mathsf{SIVP}_{\mathcal{O}(\lambda)}$ problem our scheme is fully-traceable.*

In this paper we only prove the anonymity of the scheme. Proof of traceability of the scheme is provided in the full version of this paper.

Anonymity

In the anonymity game between a challenger and an adversary, first, the challenger generates keys and gives the public keys and all the users' secret signing keys to the adversary. The adversary can query signer's index of any signature. Later, he sends two challenging indices to the challenger. The challenger selects a bit randomly from the two indices, then generates and sends back a challenging signature. The adversary wins if he can guess the index which is used to generate the challenging signature without querying.

We prove that the proposed scheme is fully anonymous using the following two games between an adversary A and a challenger C.

Game 1. In this game, the challenger C sets everything honestly. The adversary is given the group public key and the secret signing keys of all the users. The challenger answers all the opening queries that the adversary makes. Finally, the challenger produces a signature Σ^* with the true identities $(i_0, i_1, S_0, S_1, \Gamma^*, M^*)$ that the adversary sent, and forwards Σ^* to the adversary.

Game 2. In this game, instead of generating an honest non-interactive zero knowledge argument Π, the challenger simulates the argument for the challenge signature Σ^*. Thus, Game 2 is the same as Game 1 except the simulated Π^*. Since the underlying argument system is statistically zero-knowledge, the distribution of simulated Π^* is statistically close to that of the legitimate Π. Thus Game 1 and Game 2 are indistinguishable.

Indistinguishability of above two games proves that our proposed scheme is fully anonymous.

6 Conclusion

In this paper, we considered a situation where the tokens of the attributes are generated independently to the secret signing keys of the attributes to achieve full anonymity. We presented a zero-knowledge protocol that enables provers to convince the validity of them, their attributes, and the tokens in such scenarios. Moreover, we presented a new ABGS scheme with VLR from lattices to achieve full anonymity.

References

1. Bellare, M., Micciancio, D., Warinschi, B.: Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In: EUROCRYPT 2003, LNCS. vol. 2656, pp. 614–629. Springer Berlin Heidelberg (2003)
2. Chaum, D., Van Heyst, E.: Group signatures. In: EUROCRYPT 1991, LNCS. vol. 547, pp. 257–265. Springer Berlin Heidelberg (1991)
3. El Bansarkhani, R., El Kaafarani, A.: Post-quantum attribute-based signatures from lattice assumptions. IACR Cryptology ePrint Archive **2016**, 823 (2016)
4. El Kaafarani, A., Chen, L., Ghadafi, E., Davenport, J.: Attribute-based signatures with user-controlled linkability. In: Cryptology and Network Security. CANS 2014. vol. 8813, pp. 256–269. Springer, Cham (2014)
5. El Kaafarani, A., Ghadafi, E., Khader, D.: Decentralized traceable attribute-based signatures. In: CT-RSA 2014. vol. 8366, pp. 327–348. Springer, Cham (2014)
6. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: CRYPTO 1986. vol. 263, pp. 186–194. Springer (1986)
7. Gagné, M., Narayan, S., Safavi-Naini, R.: Short pairing-efficient threshold-attribute-based signature. In: Pairing 2012. vol. 7708, pp. 295–313. Springer, Berlin, Heidelberg (2012)

8. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: ACM 2008. pp. 197–206. ACM (2008)
9. Ghadafi, E.: Stronger security notions for decentralized traceable attribute-based signatures and more efficient constructions. In: CT-RSA 2015. vol. 9048, pp. 391–409. Springer, Cham (2015)
10. Herranz, J., Laguillaumie, F., Libert, B., Ràfols, C.: Short attribute-based signatures for threshold predicates. In: CT-RSA 2012, LNCS. vol. 7178, pp. 51–67. Springer, Berlin, Heidelberg (2012)
11. Ishida, A., Sakai, Y., Emura, K., Hanaoka, G., Tanaka, K.: Fully anonymous group signature with verifier-local revocation. In: Security and Cryptography for Networks. SCN 2018. vol. 11035, pp. 23–42. Springer, Cham (2018)
12. Kawachi, A., Tanaka, K., Xagawa, K.: Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: ASIACRYPT 2008, LNCS. vol. 5350, pp. 372–389. Springer Berlin Heidelberg (2008)
13. Khader, D.: Attribute based group signature with revocation. IACR Cryptology ePrint Archive **2007**, 241 (2007)
14. Khader, D.: Attribute based group signatures. IACR Cryptology ePrint Archive **2007**, 159 (2007)
15. Kuchta, V., Sahu, R.A., Sharma, G., Markowitch, O.: On new zero-knowledge arguments for attribute-based group signatures from lattices. In: ICISC 2017, LNCS. vol. 10779, pp. 284–309. Springer, Cham (2017)
16. Langlois, A., Ling, S., Nguyen, K., Wang, H.: Lattice-based group signature scheme with verifier-local revocation. In: PKC 2014, LNCS. vol. 8383, pp. 345–361. Springer Berlin Heidelberg (2014)
17. Li, J., Au, M.H., Susilo, W., Xie, D., Ren, K.: Attribute-based signature and its applications. In: Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security. pp. 60–69. ACM (2010)
18. Li, J., Kim, K.: Attribute-based ring signatures. IACR Cryptology ePrint Archive **2008**, 394 (2008)
19. Ling, S., Nguyen, K., Stehlé, D., Wang, H.: Improved zero-knowledge proofs of knowledge for the isis problem, and applications. In: PKC 2013, LNCS. vol. 7778, pp. 107–124. Springer Berlin Heidelberg (2013)
20. Ling, S., Nguyen, K., Wang, H.: Group signatures from lattices: simpler, tighter, shorter, ring-based. In: PKC 2015, LNCS. vol. 9020, pp. 427–449. Springer Berlin Heidelberg (2015)
21. Maji, H.K., Prabhakaran, M., Rosulek, M.: Attribute-based signatures. In: CT-RSA 2011, LNCS. vol. 6558, pp. 376–392. Springer Berlin Heidelberg (2011)
22. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: EUROCRYPT 2012, LNCS. vol. 7237, pp. 700–718. Springer Berlin Heidelberg (2012)
23. Perera, M.N.S., Koshiba, T.: Achieving almost-full security for lattice-based fully dynamic group signatures with verifier-local revocation. In: Information Security Practice and Experience. ISPEC 2018, LNCS. vol. 11125, pp. 229–247. Springer, Cham (2018)
24. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: In STOC. pp. 84–93. ACM Press (2005)
25. Shahandashti, S.F., Safavi-Naini, R.: Threshold attribute-based signatures and their application to anonymous credential systems. In: AFRICACRYPT 2009, LNCS. vol. 5580, pp. 198–216. Springer (2009)
26. Stern, J.: A new paradigm for public key identification. IEEE Transactions on Information Theory **42**(6), 1757–1768 (1996)

27. Zhang, Y., Gan, Y., Yin, Y., Jia, H.: Attribute-Based VLR Group Signature Scheme from Lattices: 18th International Conference, ICA3PP 2018, Guangzhou, China, November 15-17, 2018, Proceedings, Part IV, pp. 600–610 (11 2018)