Post-Quantum Attribute-Based Signatures from Lattice Assumptions

Rachid El Bansarkhani and Ali El Kaafarani

Technische Universität Darmstadt, Germany
elbansarkhani@cdc.informatik.tu-darmstadt.de
University of Oxford, UK.
Ali.ElKaafarani@maths.ox.ac.uk

Abstract. Attribute based signature schemes (ABS) constitute important and powerful primitives when it comes to protecting the privacy of the user's identity and signing information. More specifically, ABS schemes provide the advantage of anonymously signing a message once a given policy is satisfied. As opposed to other related privacy preserving signatures, the verifier is not able to deduce from the signature, which attributes have been used to satisfy the (public) signing policy. In this work we give new and efficient constructions of lattice-based ABS signature schemes, that are not based on the traditional approach of using span programs or secret sharing schemes as for classical schemes. In fact, our approach is less involved and does not require such complex subroutines. In particular, we first construct a new (t, B)-threshold ABS scheme that allows to anonymously generate signatures, if t out of $p = |B|$ attributes are covered by valid credentials. Based on this scheme, we propose a lattice-based ABS scheme for expressive $(\wedge, \vee)$-policies, by use of a new credential aggregation system that is built on top of a modified variant of Boyen's signature scheme. The signature size of the so obtained ABS scheme is linear in the number of disjunctive terms rather than the number of attributes.

Keywords: Lattice-Based Cryptography, Attribute Based Signatures

1 Introduction

Often we are less concerned with *who* signed something than with *what attributes* (e.g. director of this company) they have. We want to be able to verify the authenticity of signers without revealing attributes that can include their nationality, age, job title or any other identifying/private criterium. Attribute-Based Signatures (ABS) are a promising, versatile primitive that allows signers to *anonymously* authenticate messages while enjoying fine-grained control over identifying information, i.e. attributes. They were first introduced by Maji et al. in a preliminary version [27]. Subsequently, other ABS schemes were proposed by Li and Kim in [23], Shahandashti and Safavi-Naini in [33], and Li et al. in [22]. In an ABS scheme, users can only sign messages w.r.t. policies satisfied by a set of attributes they possess. The verifiers of a given valid ABS signature are then convinced that a signer with a set of attributes satisfying the policy in question has signed the message but learn neither the identity of the signer nor the exact attributes he/she used to produce the signature. Attribute-based signatures are a generalization of many existing and widely-used anonymous digital signature notions such as group [7] and ring [32] signatures. Attribute-based signatures have many applications including trust negotiation, e.g. [13], attribute-based messaging, e.g. [3], and leaking secrets.

Various features have been added to attribute-based signature schemes to meet real-world security requirements such as decentralization [31], traceability [12,11,16], user-controlled linkability [10], and controllable-linkability [35].

ABS schemes have different variants according to how expressive the policies they support are. For instance, we have threshold Attribute-Based Signatures (tABS), proposed by Shahandashti and Safavi-Naini [33], in which the signing policy is restricted to the threshold type, i.e. a signer who has t out of n attributes can sign a message w.r.t. the policy $\psi = (t, S)$, where $|S| = n$ for some set S of pre-specified attributes. Gagné et al. [14] gave a tABS scheme that is *pairing efficient* in the sense that they decreased the number of pairing computations. Herranz et al. [18] gave two tABS schemes with constant-size signatures. ABS schemes supporting more expressive policies, i.e. any monotonic access structure, were first given by Maji et al. [27]; Policies here take the form of *Boolean formulas*, i.e. with OR and AND gates. Okamoto and Takashima [30,31], gave constructions of ABS supporting non-monotonic access structures, i.e. negation of attributes is also allowed. Note that any scheme supporting monotonic access structures could support non-monotonic access structures simply by doubling the universe of attributes.

Other useful features were also added to ABS schemes. El Kaafarani et al. [10] recently introduced the notion of Attribute-Based Signatures with User-Controlled Linkability (ABS-UCL), which adds the user-controlled linkability feature to standard ABS schemes, where users can at discretion choose to make some of their ABS signatures, directed at a specific verifier, linkable without sacrificing their privacy.

Decentralized Traceable ABS (DTABS) schemes [11,16] are ABS schemes that don't rely on a central authority and furthermore entail traceability, which allows an opener that has a special tracing key to identify the signer of a signature in the case of misuse/dispute.

Related Work. There has been an interesting progress recently regarding quantum-resistant anonymous digital signatures. For instance, lattice-based group signatures were recently proposed in [17,20,21,26,24]. The proposed schemes improved upon the results presented by Gordon et al. (ASIACRYPT 2010), both in terms of public key and signature sizes. Later on, Boyen proposed in [5] an attribute-based functional encryption from lattices; It is in fact a key-policy attribute-based encryption scheme. However, no equivalent attribute based signature schemes for general policies from lattices have been realized so far. In [8], Cheng et al. proposed Policy-Based Signatures (PBS) from lattices, where a signer is only allowed to sign messages satisfying a certain policy, but where the

signatures do not reveal the underlying policy to the verifiers, in other words, the verifiers have no say in the policy itself. Another relevant scheme is anonymous attribute tokens (AAT) proposed by Camenish et al. in [6]. As they define it, an AAT scheme can be seen as an extension of group signatures where the issuer can assign a list of attributes to a user's signing key. The user would then need to selectively reveal some of these attributes to convince the verifier that he/she has valid credentials (i.e., signing key with attributes) certifying the claimed attribute values, but without revealing any information about the non-revealed attributes. This clearly provides a lower level of privacy compared to attribute-based signatures where signers don't need to reveal anything about their attributes except that they indeed satisfy a certain policy, i.e. verifiers can't deduce which attributes the signer has.
Due to recent developments to consider the transition to quantum resistant cryptographic primitives induced by many well-known institutions such as the National Institute of Standards and Technology (NIST), the PQCRYPTO project and the National Security Agency (NSA) we construct post-quantum secure attribute based signature schemes from lattice assumptions.

Contribution and Techniques. Our contributions are twofold. First, we construct a novel lattice-based threshold ABS scheme for a given (t, B)-policy ψ_B, where the signature size is linear in $p = |B|$. The user has to prove the possession of t valid credentials for attributes in $B \subseteq \mathsf{Attributes}$ in order to output a valid anonymous signature on a message m. However, the user does neither reveal its identity nor the attributes for which he/she has valid credentials.

1. To this end, we first modify Boyen's signature scheme that has $(\mathbf{A}, \{\mathbf{A}_i\}_{i=1}^{\ell}, \mathbf{u})$ as a public key, where $\mathbf{A}, \mathbf{A}_i \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{u} \in \mathbb{Z}_q^n$; In order to obtain a credential system for polynomially many attributes, the new scheme requires the public part to be extended to $(\mathbf{A}, \{\mathbf{A}_i\}_{i=1}^{\ell}, \mathsf{Attributes} = \{\mathbf{u}_i\}_{i=1}^{p})$. The attribute authority can now generate valid credentials for an attribute $\mathbf{u}_j$ and signer id by sampling $\mathbf{z}_{\mathsf{id},j}$ such that $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z}_{\mathsf{id},j} = \mathbf{u}_j$ and $\|\mathbf{z}_{\mathsf{id},j}\| \leq \gamma$ are satisfied. By doing so, the signer obtains valid credentials for a subset of attributes.

2. Second, and in order to anonymously prove to the verifier that a given user has indeed t valid credentials for attributes in B, we further generate fake credentials $\mathbf{d}_{\mathsf{id},j}$ for all other attributes in B such that $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{d}_{\mathsf{id},j} = \mathbf{u}_j$.

3. Third, we use a zero-knowledge system to prove the possession of t out of n attributes. Namely, we modify the statistical zero-knowledge argument of knowledge (sZKAoK) from [26]. In particular, we prove $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{x}_{\mathsf{id},j} = \mathbf{u}_j$ for $\mathbf{u}_j \in B$, where $\mathbf{x}_{\mathsf{id},j}$ is either a real or a fake credential. This is accomplished by use of masking terms for all credentials and further permutations in order to scramble the positions of the credentials in relation to the attributes, such that the verifier cannot link the credentials to the attributes when verifying the sZKAoK. Via the Fiat-Shamir transform or the post-quantum secure transformation due to Unruh [34], we obtain a signature of knowledge.

In the second part of this work, we turn the threshold ABS scheme into an ABS construction for expressive $(\wedge, \vee)$-policies. We proceed by transforming the policy into an adequate format.

1. First we turn any $(\wedge, \vee)$-policy ψ into its disjunctive normal form (DNF) such that ψ can be rewritten as

   $$\psi = C_1 \vee \ldots \vee C_k,$$

   where C_i constitutes a $\bigwedge$-policy for some attributes in B, i.e. we have $C_i = U_i^1 \wedge \ldots \wedge U_i^t$ for $i \in [k]$. The boolean variables U_i^j are set to 1, if the signer possesses valid credentials for the associated attributes $B_i = \{\mathbf{u}_{i_1}, \cdots, \mathbf{u}_{i_t}\}$, otherwise 0. Satisfying one of the disjunctive terms suffices in order to satisfy the policy, i.e. $C_i = 1 \Leftrightarrow \psi = 1$.

2. We construct from the modified Boyen signature scheme an aggregation credential system such that we can generate valid credentials for C_i if and only if the signer possesses a valid credential for each attribute related to C_i. In fact, we prove that $\mathbf{z}_{C_i} = \sum_{j=1}^{t} \mathbf{z}_{i_j}$ is an aggregate credential for the attributes in B_i, i.e.

   $$\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z}_{C_i} = \sum_{j=1}^{t} \mathbf{u}_{i_j}, \|\mathbf{z}_{C_i}\|_\infty \leq \gamma \cdot \sqrt{d_{\mathsf{max}}},$$

   where B_{max} denotes the set with the largest number of attributes $d_{\mathsf{max}} = \max_{i \in [k]} |B_i|$.

3. Subsequently, we proceed as with the threshold ABS scheme, since we can generate a valid aggregate credential for any disjunctive term C_i and fake aggregated credentials for the remaining terms. This results in a 1-out-of-k policy. Then, we can use all the same techniques as for the threshold ABS scheme. The signature size is only linear in the number k of disjunctions rather than the number of attributes $\sum_{i=1}^{k} |B_i|$.

4. Finally, we also point out how to realize traceability, which is also an important feature in order to allow the tracing authority to open signatures and trace identities in case of, for example, misbehavior or misuse of credentials. To achieve this feature, the signer has to further encrypt its identity id with the public key of the tracing authority and provide a proof for the correct format of the ciphertext. Furthermore, we present some concepts of how to extend our construction to the multi-authority setting, which reflects the situation in many real world scenarios. In fact, the threshold ABS scheme can be extended to a setting with multiple attribute authorities in a natural way. However, for expressive $(\wedge, \vee)$-policies, one has to make sure that a disjunctive term C_i is always related to attributes that are managed by one single attribute authority.

1.1 Organization

This paper is structured as follows. In Section 2, we provide the relevant background of our work. In Section 3, we present Boyen's signature scheme and our modification. Subsequently in Section 4, we give the security model of ABS schemes. In Section 5, we introduce our lattice-based threshold ABS scheme. Our zero-knowledge argument of knowledge and its features, which are applied to the threshold ABS scheme, are presented in Section 6. In Section 7 we propose our lattice-based ABS scheme for expressive policies, which is built from the threshold ABS scheme introduced in Section 5. Finally, we show in Section 8 how to use our ABS schemes within a setting involving multiple authorities.

2 Preliminaries

2.1 Notation

We denote vectors by lower-case bold letters e.g. $\mathbf{x}$, whereas for matrices we use upper-case bold letters e.g. $\mathbf{A}$. Integers modulo q are denoted by $\mathbb{Z}_q$ and reals by $\mathbb{R}$. Furthermore, we denote by $[k]$ the set of integers $\{1, \ldots, k\}$.

2.2 Discrete Gaussian Distribution

We define by $\rho : \mathbb{R}^n \to (0, 1]$ the n-dimensional Gaussian function $\rho_{s,\mathbf{c}}(\mathbf{x}) = e^{-\pi \cdot \frac{\|\mathbf{x}-\mathbf{c}\|_2^2}{s^2}}$, $\forall \mathbf{x}, \mathbf{c} \in \mathbb{R}^n$. The discrete Gaussian distribution $\mathcal{D}_{\Lambda+\mathbf{c},s}$ is defined to have support $\Lambda + \mathbf{c}$, where $\mathbf{c} \in \mathbb{R}^n$ and $\Lambda \subset \mathbb{R}^n$ is a lattice. For $\mathbf{x} \in \Lambda + c$, it basically assigns the probability $\mathcal{D}_{\Lambda+\mathbf{c},s}(\mathbf{x}) = \rho_s(\mathbf{x})/\rho_s(\Lambda + c)$.

2.3 Lattices

A k-dimensional lattice Λ is a discrete additive subgroup of $\mathbb{R}^m$ containing all integer linear combinations of k linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_k$ with $k \leq m$ and $m \geq 0$. More formally, we have $\Lambda = \{ \mathbf{B} \cdot \mathbf{x} \mid \mathbf{x} \in \mathbb{Z}^k \}$. Throughout this paper we are mostly concerned with q-ary lattices $\Lambda_q^\perp(\mathbf{A})$ and $\Lambda_q(\mathbf{A})$, where $q = \mathit{poly}(n)$ denotes a polynomially bounded modulus and $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ is an arbitrary matrix. $\Lambda_q^\perp(\mathbf{A})$ resp. $\Lambda_q(\mathbf{A})$ are defined by

$$\begin{aligned} \Lambda_q^\perp(\mathbf{A}) &= \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{A}\mathbf{x} \equiv \mathbf{0} \mod q\} \\ \Lambda_q(\mathbf{A}) &= \{\mathbf{x} \in \mathbb{Z}^m \mid \exists \mathbf{s} \in \mathbb{Z}^m \text{ s.t. } \mathbf{x} = \mathbf{A}^\top \mathbf{s} \mod q\} . \end{aligned}$$

By $\lambda_i(\Lambda)$ we denote the *i-th successive minimum*, which is the smallest radius r such there exist i linearly independent vectors of norm r (typically l_2 norm) in Λ. For instance, $\lambda_1(\Lambda) = \min_{\mathbf{x} \neq \mathbf{0}} \|\mathbf{x}\|_2$ denotes the minimum distance of a lattice determined by the length of its shortest nonzero vector.

Micciancio and Regev introduced the smoothing parameter in [29]:

Definition 1. *For any n-dimensional lattice Λ and positive real $\epsilon > 0$, the smoothing parameter $\eta_\epsilon(\Lambda)$ is the smallest real $s > 0$ such that $\rho_{1/s}(\Lambda^* \backslash \{0\}) \leq \epsilon$.*

Lemma 1. ***([15, Theorem 3.1]).*** *Let $\Lambda \subset \mathbb{R}^n$ be a lattice with basis $\mathbf{B}$, and let $\epsilon > 0$. We have*

$$\eta_\epsilon(\Lambda) \leq \| \tilde{\mathbf{B}} \| \cdot \sqrt{\ln(2n(1+1/\epsilon))/\pi}\,.$$

Specifically, we have $\eta_\epsilon(\Lambda) \leq b \cdot \sqrt{\ln(2n(1+1/\epsilon))/\pi}$ for basis $\mathbf{B} = b \cdot \mathbf{I}$ of Λ.

Lemma 2. ***([29, Lemma 4.4]).*** *Let Λ be any n-dimensional lattice. Then for any $\epsilon \in (0,1)$, $s \geq \eta_\epsilon(\Lambda)$, and $\mathbf{c} \in \mathbb{R}^n$, we have*

$$\rho_{s,\mathbf{c}}(\Lambda) \in [\frac{1-\epsilon}{1+\epsilon}, 1] \cdot \rho_s(\Lambda)\,.$$

Lemma 3 (Lemma 2.4, [2]). *For any real $s > 0$ and $T > 0$, and any $\mathbf{x} \in \mathbb{R}^n$, we have*

$$P[| \langle \mathbf{x}, \mathcal{D}_{\mathbb{Z}^n,s} \rangle | \geq T \cdot s \, \|\mathbf{x}\|] < 2exp(-\pi \cdot T^2)\,.$$

Lemma 4 ([15], Theorem 3.1). *Let $\Lambda \subset \mathbb{R}^n$ be a lattice with basis $\mathbf{S}$, and let $\epsilon > 0$.We have $\eta_\epsilon(\Lambda) \leq \| \tilde{\mathbf{S}} \| \cdot \sqrt{\ln\left(2n\left(1+\frac{1}{\epsilon}\right)\right)/\pi}$. In particular, for any function $\omega(\sqrt{\log n})$, there is a negligible $\epsilon(n)$ for which $\eta_\epsilon(\Lambda) \leq \| \tilde{\mathbf{S}} \| \cdot \omega(\sqrt{\log n})$.*

2.4 Lattice Problems.

For the SIS problem we consider the full-rank m-dimensional integer lattices $\Lambda_q^\perp(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{A}\mathbf{x} \equiv 0 \mod q\}$ consisting of all vectors that belong to the kernel of the matrix $\mathbf{A}$. In particular, $SIS_{q,n,\beta}$ is an average-case problem of the approximate shortest vector problem on $\Lambda_q^\perp(\mathbf{A})$ for $\beta > 0$. Given a uniform random matrix $\mathbf{A} \in \mathbb{Z}^{n \times m}$ with $m = poly(n)$, the problem is to find a non-zero vector $\mathbf{x} \in \Lambda_q^\perp(\mathbf{A})$ such that $\| \mathbf{x} \| < \beta$. For $q \geq \beta\sqrt{n}\omega(\sqrt{\log n})$ finding a solution to this problem is at least as hard as probabilistically $\tilde{O}(\beta\sqrt{n})$-approximating the Shortest Independent Vector Problem on n-dimensional lattices in the worst-case [15,1].

3 Boyen's Standard Model Signature Scheme

In this section we recap the construction of Boyen's signature scheme [4] that is proven to be secure in the standard model. Based on our modification, that we propose in Section 3.1, we will establish an ABS scheme, where the signature size is logarithmic in the number of users. We give a description of the underlying signature scheme instantiated with the trapdoor construction [28]. The parameters are defined as follows. Since we are signing identities later in our constructions in order to generate credentials for specific signers, we will use id as the message to be signed.

- $q = poly(n)$
- $k = \lceil \log q \rceil = O(\log n)$
- $O(1)\omega(\log n) \geq \eta_\epsilon(\Lambda^\perp(\mathbf{G}))$
- $\bar{m} = O(nk)$
- $m = \bar{m} + 2nk$
- $\mathsf{id} \in \{0,1\}^\ell$

$\mathsf{Gen}(1^n)$: Sample a matrix $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n\times\bar{m}}$ uniformly at random and select $\mathbf{T} \hookleftarrow D_{\mathbb{Z}^{\bar{m}\times nk},\omega(\sqrt{\log n})}$. Define $\mathbf{A} := [\bar{\mathbf{A}} \mid \mathbf{G} - \bar{\mathbf{A}}\mathbf{T}\ ]$ and sample uniform random matrices $\mathbf{A}_0, \mathbf{A}_i \in \mathbb{Z}_q^{n\times nk}$ for $i \in [\ell]$, where $\mathbf{G}$ is the public gadget matrix from [28]. Furthermore, select a uniform random syndrome $\mathbf{u} \in \mathbb{Z}_q^n$. Output secret key $\mathsf{sk} := \mathbf{T}$ and public key $\mathsf{pk} := (\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$.

$\mathsf{Sign}(\mathsf{sk}, \mathsf{id} \in \{0,1\}^\ell)$: Define $\mathbf{A}_{\mathsf{id}} = [\mathbf{A} \mid \sum_{i=1}^{\ell} \mathsf{id}_i \cdot \mathbf{A}_i\ ]$. Subsequently, sample a signature $\mathbf{z} \hookleftarrow D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}_{\mathsf{id}}),s}$ satisfying $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z} \equiv \mathbf{u} \bmod q$ by use of the trapdoor $\mathbf{T}$. Output $\mathbf{z}$ as the signature.

$\mathsf{Verify}(\mathsf{pk}, \mathsf{id} \in \{0,1\}^\ell, \mathbf{z})$: If $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z} \equiv \mathbf{u} \bmod q$ and $\|\mathbf{z}\| \leq s\sqrt{m}$ are satisfied, output 1, else 0.

The scheme has been proven to be secure in the standard model as long as $\mathrm{SIS}_{n,m,\beta}$ is hard to solve for $\beta = O(\ell(nk)^{3/2}) \cdot \omega(\sqrt{\log n})^3$.

3.1 Modification of Boyen's Signature Scheme

In our construction, we let the attribute authority apply a variant of Boyen's signature scheme from lattice assumptions. It is closely related to the pairing-based signature scheme of Waters [36]. However, for our anonymous protocol we hide some public parts of the signature and prove in zero-knowledge the possession of valid signatures. To this end, the identities of the users are encoded as messages and both the identity and signatures represent the secret credentials of a user. In order to allow for various attributes, we modify Boyen's signature scheme in such a way that we extend the number $p = poly(n)$ of publicly available syndromes.

$\mathsf{Gen}(1^n)$: Sample a matrix $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n\times\bar{m}}$ uniformly at random and select $\mathbf{T} \hookleftarrow D_{\mathbb{Z}^{\bar{m}\times nk},\omega(\sqrt{\log n})}$. Define $\mathbf{A} := [\bar{\mathbf{A}} \mid \mathbf{G} - \bar{\mathbf{A}}\mathbf{T}\ ]$ and sample uniform random matrices $\mathbf{A}_i \in \mathbb{Z}_q^{n\times nk}$ for $i \in [\ell]$. Furthermore, select a set of uniform random syndromes $\mathbf{u}_1, \ldots, \mathbf{u}_p \in \mathbb{Z}_q^n$. Output secret key $\mathsf{sk} := \mathbf{T}$ and public key $\mathsf{pk} := (\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathsf{Attributes} = \{\mathbf{u}_i\}_{i=1}^p)$.

$\mathsf{Sign}(\mathsf{sk}, \mathsf{id} \in \{0,1\}^\ell, \mathbf{u} \in \mathsf{Attributes})$: Define $\mathbf{A}_{\mathsf{id}} = [\mathbf{A} \mid \sum_{i=1}^{\ell} \mathsf{id}_i \cdot \mathbf{A}_i\ ]$. Subsequently, sample a signature $\mathbf{z} \hookleftarrow D_{\Lambda_{\mathbf{u}}^\perp(\mathbf{A}_{\mathsf{id}}),s}$ satisfying $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z} \equiv \mathbf{u} \bmod q$ by use of the trapdoor $\mathbf{T}$. Output $\mathbf{z}$ as the signature.

$\mathsf{Verify}(\mathsf{pk}, \mathsf{id} \in \{0,1\}^\ell, \mathbf{u} \in \mathsf{Attributes}, \mathbf{z})$: If $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z} \equiv \mathbf{u} \bmod q$ and $\|\mathbf{z}\| \leq s\sqrt{m}$ are satisfied, output 1, else 0.

By a straightforward proof, the modified scheme is shown to be still secure in the standard model as long as $\mathrm{SIS}_{n,m,\beta}$ is hard to solve for $\beta = O(\ell(nk)^{3/2}) \cdot \omega(\sqrt{\log n})^3$.

Theorem 1 (Adapted [28]). *There exists a* PPT *oracle algorithm (a reduction)* $\mathcal{S}$ *attacking the problem for large enough* $\beta = O(\ell(nk)^{3/2}) \cdot \omega(\sqrt{\log n})^3$ *such that, for any adversary* $\mathcal{F}$ *mounting an SU-CMA attack (strongly unforgeable under chosen message attacks) on the signature scheme* BS^* *above with* p *syndromes and making at most* Q *queries,*

$$\mathsf{Adv}_{SIS_{n,m,\beta}}(\mathcal{S}^{\mathcal{F}}) \leq \frac{1}{p}\mathsf{Adv}_{\mathsf{BS}^*}^{SU\text{-}CMA}(\mathcal{F})/(2(\ell-1)Q+2) - \mathsf{negl}(n)$$

Proof. The proof of this theorem is straightforward strictly following the proof steps of [28]. The only difference is that the adversary is now given the choice to provide a forgery for any of the $\mathbf{u} \in \mathsf{Attributes}$. So the advantage of the adversary is higher by a factor of $poly(n)$, which is still negligible. In particular, the reduction $\mathcal{S}$ obtains $\bar{m} + nk + p$ uniform random vectors in $\mathbb{Z}_q^n$ as input and parses them as a matrix $\mathbf{A} = [\ \bar{\mathbf{A}} \mid \mathbf{B}\ ] \in \mathbb{Z}_q^{n\times(\bar{m}+nk)}$ and p syndromes $\mathbf{u}_1', \ldots, \mathbf{u}_p'$. The reduction will then use the adversary $\mathcal{F}$ to find $\mathbf{z} \in \mathbb{Z}^m$ with $\|\mathbf{z}\| \leq \beta - 1$ such that $\mathbf{A}\mathbf{z} = \mathbf{u}_i \bmod q$ for any $i \in [p]$, i.e. $[\ \mathbf{A} \mid \mathbf{u}_1 \mid \ldots \mid \mathbf{u}_p\ ]\cdot \begin{bmatrix} \mathbf{z} \\ -\mathbf{e}_i \end{bmatrix} = \mathbf{0} \bmod q$, where $\mathbf{e}_i \in \mathbb{Z}^p$ is the i-the unit vector with 1 at position i and zero elsewhere. Or $\mathcal{S}$ will use $\mathcal{F}$ to generate $\mathbf{z} \in \mathbb{Z}^m \backslash \{\mathbf{0}\}$ such that $[\ \mathbf{A} \mid \mathbf{u}_1 \mid \ldots \mid \mathbf{u}_p\ ]\cdot \begin{bmatrix} \mathbf{z} \\ \mathbf{0} \end{bmatrix} = \mathbf{0} \bmod q$. All other proof steps of [28] essentially remain the same. □

4 Definition and Security Model

In this section, we define the syntax and security of Attribute-Based Signatures.

4.1 Syntax of ABS

An ABS scheme consists of the following algorithms [27]:

- $\mathsf{Setup}(1^\lambda)$: On input a security parameter λ, it returns public parameters pp.
- $\mathsf{AASetup}(\mathsf{pp}, \mathsf{aid})$: It is run by attribute authority $\mathsf{AA}_{\mathsf{aid}}$ to generate its public/secret key pair $(\mathsf{vk}_{\mathsf{AA}}, \mathsf{sk}_{\mathsf{AA}})$. The authority publishes $\mathsf{vk}_{\mathsf{AA}}$ and keeps $\mathsf{sk}_{\mathsf{AA}}$ secret.
- $\mathsf{AttKeyGen}(\mathsf{pp}, \mathsf{id}, \mathsf{at}, \mathsf{sk}_{\mathsf{AA}})$: It is run by attribute authority AA who manages attribute at, it gives the user id the secret key $\mathsf{sk}_{\mathsf{id},\mathsf{at}}$.
- $\mathsf{Sign}(\mathsf{pp}, m, \psi, \mathsf{sk}_{\mathsf{id},\mathcal{A}})$: It takes the message m, the policy ψ, and the signer's secret key $\mathsf{sk}_{\mathsf{id},\mathcal{A}}$ and outputs $\perp$ if $\psi(\mathcal{A}) = 0$, otherwise, it returns a signature σ.
- $\mathsf{Verify}(\mathsf{pp}, \sigma, \psi, \{\mathsf{vk}_{\mathsf{AA}_i}\}_i, m)$: It takes a signature σ, a message m, the signing policy ψ and the public keys of attribute authorities involved in ψ, i.e. $\{\mathsf{vk}_{\mathsf{AA}_i}\}_i$. It returns 1 if σ is a valid signature on the message m w.r.t. the signing policy ψ and 0 otherwise.

4.2 Security Requirements

Besides correctness, the security of an ABS scheme requires:

Unforgeability. This requires that users cannot output signatures on messages w.r.t a signing policy not satisfied by their set of attributes, even if they pool their attributes together, which ensures resistance against collusion.

Definition 2 (Unforgeability). *An* ABS *scheme is unforgeable if for all security parameters* $\lambda \in \mathbb{N}$*, for all PPT adversaries the advantage in winning the following game is negligible:*

Setup: *The challenger runs* Setup *and gives* pp *to the adversary.*

Play: *Throughout the game the adversary can ask for attribute authorities to be created and ask for their secret keys. He can also ask for honest users to be created and ask for their personal secret keys. He can also ask for keys for attributes for users and signatures on tuples* (m, ψ) *of his choice on behalf of honest users.*

Output: *The adversary outputs* (σ^*, m^*, ψ^*)

We say that the adversary wins the game if σ^* *is a valid signature on* m^* *w.r.t.* ψ^**, where* (m^*, ψ^*) *was not queried to the signing oracle, and there exists no subset of attributes* $\mathcal{A}^*$ *whose keys have been revealed to the adversary or managed by corrupt attribute authorities s.t.* $\psi(\mathcal{A}^*) = 1$*.*

Anonymity. This requires that a signature reveals neither the identity of the signer nor the attributes used in the signing engine.

Definition 3 (Anonymity). *An* ABS *scheme is anonymous if for all security parameters* $\lambda \in \mathbb{N}$*, for all PPT adversaries the advantage in winning the following game is negligible:*

Setup: *The challenger runs* Setup *and gives* pp *to the adversary.*

Phase I: *The adversary can fully control* all *attribute authorities. It can also ask for the secret keys of signers of his choice; those signers will be referred to as corrupt users. Also, the adversary can ask for the secret key of any attribute and signatures on tuples* (m, ψ) *of his choice on behalf of honest users.*

Challenge: *The adversary outputs* $(m, \mathsf{id}_0, \mathcal{A}_0, \mathsf{id}_1, \mathcal{A}_1, \psi)$ *where* $\psi(\mathcal{A}_i) = 1$ *for* $i = 0, 1$*. We require that both* id_0 *and* id_1 *are honest users. The adversary gets back a signature* σ_b *produced using* $(\mathsf{id}_b, \mathcal{A}_b)$ *for* $b \leftarrow \{0, 1\}$*.*

Phase II: *Same as in phase I with the additional condition that the adversary cannot corrupt any of* id_0 *or* id_1*.*

Output: *The adversary outputs its guess* b^* *and wins if* $b^* = b$*.*

More formally, we define the advantage of an adversary $\mathcal{F}$ in winning the anonymity game as

$$\mathsf{Adv}^{\text{anon}}_{\mathcal{F},\text{ABS}}(\lambda) = |\Pr[b^* = b] - 1/2|$$

5 Threshold Attribute-based Signatures

In this section, we will present a new threshold ABS scheme from lattice assumptions that is built on top of the signature scheme introduced in Section 3.1. It turns out that this construction can efficiently be turned into an attribute based signature scheme. In particular, we let the public part $\mathbf{u}$ of $(\mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u})$ take over the role of the public attributes. The attribute authority generates arbitrary many uniform random elements $\mathbf{u}_i$, which together represent all available attributes, such that a particular user id is assigned a number of attributes $\mathbf{u}_i$ if he possesses valid signatures $\mathbf{z}_i$ on those attributes $\mathbf{u}_i$, i.e. $\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z}_i = \mathbf{u}_i \bmod q$ and $\|\mathbf{z}_i\| \leq \gamma$. Thus, the public key and attributes are given by the tuple $(\mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathsf{Attributes} = \{\mathbf{u}_i\}_{i=1}^{p})$.

5.1 Construction

$\mathsf{Setup}_\mathsf{T}(1^\lambda)$: The public parameters are set to n and q and the discrete Gaussian parameter s.

$\mathsf{AASetup}_\mathsf{T}(\mathsf{pp}, \mathsf{aid})$: The attribute authority generates public random matrices $\mathsf{vk}_\mathsf{AA} := \{\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n \times m}\}$ as the public key and the associated trapdoor $\mathsf{sk}_\mathsf{AA} := \mathbf{T}$ according to the modified Boyen's signature scheme in Section 3.1, where 2^ℓ denotes the number of users. Furthermore he generates a set of attributes

$$\mathsf{Attributes} = \{\mathbf{u}_1, \ldots, \mathbf{u}_p\},$$

where $\mathbf{u}_i \in \mathbb{Z}_q^n$ is uniform random. Each attribute at_i is associated to a uniform random element $\mathbf{u}_i$, for instance via a public list of tuples $Q[i] = (\mathsf{at}_i, \mathbf{u}_i)$.

$\mathsf{AttKeyGen}_\mathsf{T}(\mathsf{pp}, \mathsf{id}, B \subseteq \mathsf{Attributes}, \mathsf{sk}_\mathsf{AA})$: A certain user represented as a bit string $\mathsf{id} \in \{0,1\}^\ell$ is assigned a set of attributes

$$\mathsf{sk}_\mathsf{id} = \{(\mathbf{z}_{\mathsf{id},j_1}, \mathbf{u}_{j_1}), \ldots, (\mathbf{z}_{\mathsf{id},j_k}, \mathbf{u}_{j_k})\}$$

by the attribute authority such that

$$\mathbf{A}_\mathsf{id} \cdot \mathbf{z}_{\mathsf{id},j_1} = \mathbf{u}_{j_1} \bmod q,\ \ \|\mathbf{z}_{\mathsf{id},j_1}\| \leq \gamma$$

is satisfied for $\mathbf{u}_j \in B$.

$\mathsf{Sign}_\mathsf{T}(\mathsf{pp}, m, \psi, \mathsf{sk}_\mathsf{id})$: On input the message m, a policy ψ and the secret key, the signer generates a signature of knowledge

$$\begin{aligned} \mathit{\Pi} = \mathsf{SPoK}(&\mathsf{public} := \{m, \mathsf{Attributes}, \mathbf{A}\},\ \mathsf{witness} := \{\mathsf{sk}_{id}\} : \\ &\exists\ B \subseteq \mathsf{sk}_\mathsf{id} \text{ s.th. } \psi(B) = 1) \end{aligned}$$

Output signature $\mathit{\Sigma} = (m, \mathit{\Pi})$.

$\mathsf{Verify}_\mathsf{T}(\mathsf{pp}, \Sigma, \psi, \mathsf{vk}_\mathsf{AA}, \mathsf{Attributes})$: On input the policy, the list of attributes, the public verification matrices vk_AA and an ABS signature Σ, which is parsed as $\Sigma = (m, \Pi)$, the verifier returns 1 if $\Pi = \mathsf{SPoK}$ is a valid proof, otherwise he outputs 0.

We note that a user cannot generate signatures on an attribute unless he can solve SIS instances. And different signers cannot collude in order to generate a signature on an attribute due to differing public keys $\mathbf{A}_\mathsf{id}$. This directly follows from the unforgeability of the underlying signature scheme.

At a high level view of our threshold ABS scheme and for a given policy $\psi_{A,t}$ with $A \subseteq \mathsf{Attributes}$, a user id satisfying this policy such that $\psi_{A,t}(\mathsf{sk}_\mathsf{id}) = 1$ utilizes t signatures for attributes in $B \subseteq A$ and generates $|A| - t$ "dummy" signatures on the remaining ones in $A \backslash B$. In the non-interactive zero-knowledge argument of knowledge, we hide the relationship between the signatures and the attributes. This is done by use of different permutations shuffling the positions, when constructing the commitments, such that the verifier cannot link any signature to a particular attribute. He will only be able to observe the sizes and that t out of the signatures are valid.

We now present a description of the SPoK used in our construction. More specifically, we modify the Stern-like statistical zero-knowledge argument system that has been applied in [26] for group signature schemes. Following this the soundness property is guaranteed for computationally bounded cheating provers and the zero-knowledge property is satisfied even for unbounded cheating verifiers. We realize the statistical zero-knowledge argument system by use of the statistical hiding commitment scheme of Kawachi et al. [19], where the binding property (computational) is based on the worst-case hardness of $\mathsf{SIVP}_{\tilde{O}(n)}$.

The sZKAoK construction from [26] is designed to prove the knowledge of a message signature pair $(\mathsf{id}, \mathbf{z}) \in \{0,1\}^\ell \times \mathbb{Z}^{2m}$ for Boyen's signature scheme such that $\|\mathbf{z}\|_\infty \leq \gamma$ and $\mathbf{A}_\mathsf{id} \cdot \mathbf{z} \equiv \mathbf{u} \bmod q$. The public key is given by $\mathsf{vk}_\mathsf{AA} := \{\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell) \in \mathbb{Z}_q^{n \times m}\}$, where $\mathbf{A}_\mathsf{id} = [\mathbf{A} | \mathbf{A}_0 + \sum_{i=1}^{\ell} \mathsf{id}_i \mathbf{A}_i]$. To have a unique public key for all users the public matrix $\bar{\mathbf{A}} = [\mathbf{A} | \mathbf{A}_0 | \ldots | \mathbf{A}_\ell] \in \mathbb{Z}_q^{n \times (2+\ell)m}$ has been introduced such that

$$\mathbf{A}_\mathsf{id} \cdot \mathbf{z} = \bar{\mathbf{A}} \cdot f_\mathsf{id}\left(\mathbf{z}\right) \equiv \mathbf{u} \bmod q$$

$$f_\mathsf{id}(\mathbf{z}) = f_\mathsf{id}\left(\mathbf{z}^{(1)}, \mathbf{z}^{(2)}\right) = \left(\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \mathsf{id}_1 \cdot \mathbf{z}^{(2)}, \ldots, \mathsf{id}_\ell \cdot \mathbf{z}^{(2)}\right)^\top \in \mathbb{Z}^{(2+\ell)m}.$$

Now, the goal is to prove in zero-knowledge that $\bar{\mathbf{A}} \cdot f(\mathbf{z}, \mathsf{id}) \equiv \mathbf{u} \bmod \ q$, $\|\mathbf{f}_\mathsf{id}(\mathbf{z})\| \leq \gamma$ and $\mathbf{f}_\mathsf{id}(\mathbf{z})$ is of the form as described above, where the entries are either set to zero or $\mathbf{z}^{(2)}$ according to the bit positions of id.

- To this end, the identifier has been extended to a bit vector $\mathsf{id}^* = (\mathsf{id}_1, \ldots, \mathsf{id}_{2\ell}) \in \mathsf{B}_{2\ell}$ having Hamming weight ℓ for every user. Permutations are applied to finally scramble the real structure of id^* such that the verifier still knows that a valid identity is concealed due to the constant Hamming weight. But he cannot correctly guess which identity has been scrambled.

- Proving that $\|f_{\mathsf{id}}(\mathbf{z})\| \leq \gamma$ is accomplished by use of the Decomposition-Extension Technique introduced in [25]. In particular, the signature parts $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}$ are split into $k = \lfloor \log \gamma \rfloor + 1$ vectors $\mathbf{z}_i^{(1)}, \mathbf{z}_i^{(2)} \in \mathbb{Z}^m,\ 1 \leq i \leq k$ of similar shape with entries in $\{-1, 0, 1\}$ such that

$$\mathbf{z} = \sum_{i=1}^{k} \gamma_i \cdot (\mathbf{z}_i^{(1)}, \mathbf{z}_i^{(2)})^\top \text{ for}$$

$$\gamma_1 = \lceil \gamma/2 \rceil, \gamma_i = \lceil (\gamma - \sum_{j=0}^{i-1} \gamma_j)/2 \rceil, \gamma_k = 1,\ 1 < i < k\,.$$

  The vectors $\mathbf{z}_i^{(1)}, \mathbf{z}_i^{(2)}$ are subsequently extended to vectors $\tilde{\mathbf{z}}_i^{(1)}, \tilde{\mathbf{z}}_i^{(2)} \in \mathsf{B}_{3m}$ such that each vector contains exactly m entries of each of $\{-1, 0, 1\}$. Subsequently, we have

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{i=1}^{k} \gamma_i \cdot f_{\mathsf{id}^*} \left(\tilde{\mathbf{z}}_i^{(1)}, \tilde{\mathbf{z}}_i^{(2)} \right) \right) = \bar{\mathbf{A}} \cdot f_{\mathsf{id}}(\mathbf{z}) \equiv \mathbf{u} \bmod q \text{ for}$$

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

  Similar to the case of id^*, where permutations are applied to scramble the structure, we use permutations to hide the structure of $\mathbf{z}_i$ as well.

Using these tools we show how to scramble and mask the different secret vectors.

- **Scrambling**: All secret vectors $(\mathbf{z}, \mathsf{id})$ are extended into vectors either with the same Hamming weight or same number of entries from each of $\{-1, 0, 1\}$. By use of permutations in a Stern-like fashion the real structure is concealed in such a way that without the knowledge of the permutations each identity could have been the real signer and each vector in B_{3m} could have been the preimage of the permuteted vectors. We now give an overview of how to apply the permutations. Let $\pi, \phi \in \mathsf{S}_{3m}$ and $\tau \in \mathsf{S}_{2\ell}$, where $\tau(\mathsf{id}^*)$ represents a permutation on the bits of the extended identity id^*. Furthermore, define the permutation

$$P_{\pi,\phi,\tau}(\mathbf{y}) = (\pi(\mathbf{x}), \phi(\mathbf{y}_0), \phi(\mathbf{y}_{\tau(1)}), \ldots, \phi(\mathbf{y}_{\tau(2\ell)})),$$

  which reorders the blocks $\mathbf{y}_i \in \mathbb{Z}^{3m}$ for $1 \leq i \leq 2\ell$ and shuffles each block with either ϕ or π. In order to convince the verifier that $f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i)$ is related to id^* for $\tilde{\mathbf{z}}_i = (\tilde{\mathbf{z}}_i^{(1)}, \tilde{\mathbf{z}}_i^{(2)})$, the verifier checks for all $1 \leq j \leq k$ that $\mathbf{y}^j = P_{\pi,\phi,\tau}(f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_j))$ is a valid vector for $\mathbf{v} = \tau(\mathsf{id}^*)$, i.e. we obtain $\mathbf{y}^j = (\mathbf{x}^j, \mathbf{y}_0^j, v_1 \cdot \mathbf{y}_0^j, \ldots, v_{2\ell} \cdot \mathbf{y}_0^j)$.

- **Masking**: In order to mask the signature parts $f_{\mathsf{id}}(\tilde{\mathbf{z}}_i)$ for $1 \leq j \leq k$ the signer samples uniform random elements $\mathbf{r}^j \hookleftarrow\in \mathbb{Z}_q^{(2+2\ell)3m}$ such that the verifier can check the relation $\mathbf{A}_{\mathsf{ext}} \cdot (\sum_{i=1}^{k} \gamma_i \cdot f_{\mathsf{id}}(\tilde{\mathbf{z}}_i)) \equiv \mathbf{u} \bmod q$ via

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{i=1}^{k} \gamma_i \cdot f_{\mathsf{id}}\left(\tilde{\mathbf{z}}_i\right) + \mathbf{r}^i \right) - \mathbf{u} \equiv \mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{i=1}^{k} \gamma_i \cdot \mathbf{r}^{(i)} \right)\,.$$

The prover ensures, when building the commitments and responses for the $\mathsf{sNIZKoA}$ proof, that the structure of the secret key material is concealed either by the scrambling or hiding techniques. This prevents the verifier from learning sensible information out of the proof.

5.2 Ingredients of our Threshold ABS scheme

In the following section we show how to extend the previous construction in order to realize a threshold ABS scheme from lattice assumptions. More precisely, we introduce further permutations and "dummies" to hide the attributes, for which the user possesses valid signatures. Based on the basic scheme, we use 2 further techniques in order to achieve anonymity of the attributes.

- **Generation of Fake Credentials**: Suppose the user id possesses valid credentials for the attributes $C = \{\mathbf{u}_{j_1}, \ldots, \mathbf{u}_{j_k}\}$ and the policy is given by $\psi = \{t \text{ out of } \mathsf{B} \subseteq \mathsf{Attributes}, t \in \mathbb{N} \wedge (B = \{\mathbf{u}_{i_1}, \ldots, \mathbf{u}_{i_{\bar{p}}}\})\}$, then the user defines two sets $A_1 = B \cap C$ and $A_2 = B \backslash A_1$. Furthermore, assume that $|A_1| = t$ and hence $|A_2| = \bar{p} - t$.

  1. **Real Credentials**: For A_1 the user applies the associated secret credentials from $\mathsf{sk}_{\mathsf{id}}$ and generates using the Decomposition-Extension technique t sets of vectors $\{\tilde{\mathbf{z}}_i^j \in \mathsf{B}_{3m}\}_{j=1}^k$ for $i \in [t]$ such that

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i^j) \right) = \bar{\mathbf{A}} \cdot f_{\mathsf{id}}(\mathbf{z}_i) \equiv \mathbf{u}_i \in A_1,\ i \in [t]\,.$$

     For each set of vectors, we generate masking terms $\{\mathbf{r}_i^j\}_{j=1}^k,\ i \in [t]$ as described above, such that the verifier can check

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \left(\mathbf{r}_i^j + f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i^j) \right) \right) - \mathbf{u}_i = \mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \mathbf{r}_i^j \right) \quad \text{for } \mathbf{u}_i \in A_1$$

  2. **Fake Credentials**: For the remaining attributes in A_2 the user generates $\bar{p} - t$ uniform random sets of dummy variables $\{\mathbf{d}_i^j \in \mathbb{Z}_q^{(2+\ell)3m}\}_{j=1}^{\bar{p}-t}$ such that

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \mathbf{d}_i^j \right) \equiv \mathbf{u} \in A_2,\ t < i \leq n$$

     Similar to the case with real credentials, the signer samples $\{\mathbf{r}_i^j\}_{j=1}^k$ for $t < i \leq n$ such that

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \left(\mathbf{r}_i^j + \mathbf{d}_i^j \right) \right) - \mathbf{u}_i = \mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \mathbf{r}_i^j \right) \quad \text{for } \mathbf{u}_i \in A_2\,.$$

- **Scrambling**: We apply permutations $P_{\pi_j,\phi_j,\tau}(\cdot)$ to all sets of vectors. We show in Section 6 how to reduce the $\mathsf{sNIZAoK}$ related cost of communication. In particular, all the permutations are completely revealed when the challenge is $CH = 2, 3$. Via a large enough seed, the verifier can recover the respective permutations.
  In our construction we require one further permutations $\xi \in \mathsf{S}_n$, which has the goal to scramble the positions of the attributes within the commitments. This prevents the verifier from learning anything about the target attributes in A_1, when viewing the set of vectors $\{P_{\pi_j,\phi_j,\tau}(f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i^j))\}_{j=1}^k$ for $i \in [t]$ with small entries in the response part of the signature. This is possible, since he cannot link these vectors with the attributes. For the sake of illustration, we give a very rough sketch of the main changes within the commitments ignoring all other parts. Let $\mathbf{r}_i = \left(\sum_{j=1}^k \gamma_i \cdot \mathbf{r}_i^j\right)$. Furthermore, denote by $P_j(\cdot)$ some permutation and denote the real/fake credentials by $\mathbf{x}_i^j = f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i^j)$ for $i \in [t]$ and $\mathbf{x}_i^j = \mathbf{d}_i^j$ for $t < i \leq n$. Then, the commitments $\mathbf{c}_i$ are of the following form, where COM denotes a statistically hiding and computationally binding commitment scheme.

$$\mathbf{c}_1 = \mathsf{COM}(\ldots\ [\ \mathbf{A}_{\mathsf{ext}} \cdot \mathbf{r}_1, \ldots, \mathbf{A}_{\mathsf{ext}} \cdot \mathbf{r}_p],\ \xi,\ \ldots)$$

$$\mathbf{c}_2 = \mathsf{COM}(\{P_j(\mathbf{r}_{\xi(1)}^j)\}_{j=1}^k,\ \ldots\ ,\ \{P_{(n-1)k+j}(\mathbf{r}_{\xi(p)}^j)\}_{j=1}^k,\ \ldots)$$

$$\mathbf{c}_3 = \mathsf{COM}(\{P_j(\mathbf{x}_{\xi(1)}^j + \mathbf{r}_{\xi(1)}^j)\}_{j=1}^k,\ \ldots\ ,\ \{P_{(n-1)k+j}(\mathbf{x}_{\xi(p)}^j + \mathbf{r}_{\xi(p)}^j)\}_{j=1}^k,\ \ldots)$$

  By use of the additional permutation ξ, the verifier is prevented from learning, for which attributes the user possesses real credentials. Intuitively, this follows from the fact that when $CH = 1$, the prover has to build a response such that the verifier can check $\mathbf{c}_1$ and $\mathbf{c}_2$. Only in this case, the prover reveals small elements, i.e. he provides permuted set of vectors

$$\{\{P_j(\mathbf{r}_{\xi(1)}^j)\}_{j=1}^k, \{P_j(\mathbf{x}_{\xi(1)}^j)\}_{j=1}^k, \ldots, \{P_j(\mathbf{r}_{\xi(1)}^j)\}_{j=1}^k, \{P_j(\mathbf{x}_{\xi(1)}^j)\}_{j=1}^k\},$$

  where the verifier sees permuted vectors $\mathbf{x}_{\xi(1)}^j$ of small elements and neither learns the permutation ξ itself nor the relationship to the attributes.

  Example 1. Suppose we have 5 attributes $\mathsf{Attributes} = \{\mathbf{u}_1, \ldots, \mathbf{u}_5\}$ and user id has 2 real credentials $\mathbf{z}_1$ and $\mathbf{z}_3$. Suppose the policy is given by $\psi = \{2$ **out of** 5 **attributes**$\}$. For the sZKAoK the prover generates a uniform random permutation such as $\xi = (\ 3\ 5\ 1\ 4\ 2)$, which leads to the following response for $CH = 1$:

$$\{\{\bar{\mathbf{r}}_3^j, \bar{\mathbf{x}}_3^j\}_{j=1}^k, \{\bar{\mathbf{r}}_5^j, \bar{\mathbf{x}}_5^j\}_{j=1}^k, \{\bar{\mathbf{r}}_1^j, \bar{\mathbf{x}}_1^j\}_{j=1}^k, \{\bar{\mathbf{r}}_4^j, \bar{\mathbf{x}}_4^j\}_{j=1}^k, \{\bar{\mathbf{r}}_2^j, \bar{\mathbf{x}}_2^j\}_{j=1}^k\},$$

  where $\bar{\mathbf{r}}_i^j$ denote the permuted and processed masking terms and $\bar{\mathbf{x}}_i^j$ the scrambled fake/real credentials, out of which 2 contain small elements.

In case of $CH = 2, 3$ the verifier indeed obtains all permutations but he only views random vectors with no relationship to the real credentials, i.e. he either obtains $\{P_j(\mathbf{r}^j_{\xi(i)})\}_{j=1}^k$ or $\{P_j(\mathbf{r}^j_{\xi(i)} + \mathbf{x}^j_{\xi(i)})\}_{j=1}^k$.

A full description of the protocol is given below in Section 6. In the following section we prove the security of the scheme.

5.3 Security Proofs

Theorem 2 (Unforgeability). *If the non-interactive zero-knowledge (NIZK) system has special soundness and the modified Boyen's signature scheme is unforgeable, then our threshold ABS scheme is unforgeable in the random oracle model.*

Proof. We show that if an adversary $\mathcal{C}$ against the unforgeability of the ABS exists, then we can construct an adversary $\mathcal{F}_1$ against the soundness of the NIZK system and and adversary $\mathcal{F}_2$ against the unforgeability property of the modified Boyen's signature scheme (MBSS) for which we have:

$$\mathsf{Adv}^{\text{Unforg}}_{\mathsf{ABS},\mathcal{C}}(\lambda) \leq \mathsf{Adv}^{\text{sound}}_{\mathsf{NIZK},\mathcal{F}_1}(\lambda) + \mathsf{Adv}^{\text{Unforg}}_{\mathsf{MBSS},\mathcal{F}_2}(\lambda)$$

First, by the soundness of the NIZK system, the adversary has negligible probability to successfully fake proofs for false statements. Second, we will show how to reduce the unforgeability of our ABS scheme to the unforgeability of the modifed Boyen scheme. The adversary $\mathcal{F}_2$ will first get the verification key of the modified Boyen signature scheme $\mathsf{vk} = (\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \{\mathbf{u_i}\}_{i=1}^p)$ from its game and have access to a signing oracle to obtain signature's on identities/attributes of his choice. It then forwards $\mathsf{pp} = \{\mathsf{vk}, \mathcal{H}\}$ to $\mathcal{C}$, where $\mathcal{H}$ is a hash function that is modeled as a random oracle.

The adversary $\mathcal{F}_2$ can answer all key generation queries by simply forwarding them to its Sign oracle. When asked for signing queries, it can simulate the NIZK proofs and forward them to $\mathcal{C}$. Eventually, $\mathcal{C}$ outputs his forgery; by the extractability property of the NIZK, the adversary $\mathcal{F}_2$ can extract the witness which consists of a set of $\{\mathbf{z}_i\}_i$, where at least one of these signatures was not obtained from the signing oracle. The adversary $\mathcal{F}_2$ then forwards this signature as its forgery to its game. □

Theorem 3 (Anonymity). *If the NIZK system has statistical zero-knowledge, then our threshold ABS scheme is anonymous in the random oracle model.*

Proof. We will show that the adversary can't distinguish between the following two games but with negligible probability.

Game 1. In this game, the challenger sets up everything honestly, i.e. he generates the $(\mathsf{vk}, \mathsf{sk})$ of the modified Boyen scheme and publishes vk. Therefore, he can answer all the key generation and signing queries sent by the adversary. When he receives the challenge query with honest identities $(\mathsf{id}_0, \mathsf{id}_1, m, \mathcal{A}_0, \mathcal{A}_1, \psi)$, he chooses the bit $b \leftarrow \{0,1\}$ at random, produces an ABS signature σ_b and forwards it to the adversary.

Game 2. In this game, the challenger does exactly the same things except that when asked to respond to the challenge query, he simulated a proof that doesn't involve a witness in it, and therefore is independent of the bit b.

It is easy to see that, by the statistical zero-knowledge property of the NIZK system, these two games are indistinguishable and therefore our ABS scheme is anonymous. □

6 Zero-Knowledge Argument of Knowledge

We now give a full description of the sZKAoK proof system based on the techniques introduced in Section 5. We note that the protocol can be made non-interactive via the Fiat-Shamir heuristic, where the challenge is computed as $\{CH_j\}_{j=1}^{c} = \mathcal{H}(m, \{CMT_j\}_{j=1}^{c}, \mathsf{pp}, \mathsf{vk}_{\mathsf{AA}})$. In order to obtain a post-quantum secure non-interactive protocol we can use the transformation of Unruh [34], which is secure in the quantum random oracle model. Though there exist other approaches [9] to transform some protocols with oblivious commitments into a post-quantum secure setting, the transformation due to Unruh is universal and can be applied to any sigma protocol that entails the standard properties honest verifier zero-knowledge and special soundness. For the sake of simplicity, we mainly consider the Fiat-Shamir transform throughout this work in order to illustrate how our constructions work.
Thus, let $k = \lfloor \log \gamma \rfloor + 1$ and n the number of attributes in the policy ψ. Furthermore, denote by $\mathbf{x}_i^j = f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i^j)$ for $1 \leq i \leq t$ the decomposition of the real credentials for attributes in A_1 and by $\mathbf{x}_i^j = \mathbf{d}_i^j$ for $t < i \leq p$ the decomposition of the fake credentials for attributes in A_2. We modify the sZAoK of [26]. As a commitment scheme, we can use the one proposed by Kawachi et al. [19].

Commitments

- Generate masking terms $\{\mathbf{r}_i^j \hookleftarrow \mathbb{Z}_q^{(2\ell+2)3m}\}_{j=1}^{k}$ for $i \in [p]$, $j \in [k]$ and $\mathbf{r}_{\mathsf{id}^*} \hookleftarrow \mathbb{Z}_q^{2\ell}$
- Sample permutations $\tau \hookleftarrow \mathsf{S}_{2\ell}, \{\phi_j \hookleftarrow \mathsf{S}_{3m}\}_{j=1}^{p\cdot k}, \{\pi_j \hookleftarrow \mathsf{S}_{3m}\}_{j=1}^{p\cdot k}$ (for each attribute k permutations), $\xi \hookleftarrow \mathsf{S}_n$

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

- $\mathbf{c}_1 = \mathsf{COM}([\ \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{r}_1^j), \ldots, \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{r}_p^j)],\ \tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi)$
- $\mathbf{c}_2 = \mathsf{COM}(\{P_{\pi_j,\phi_j,\tau}(\mathbf{r}_{\xi(1)}^j)\}_{j=1}^{k},\ \ldots\ ,\ \{P_{\pi_{s_n+j},\phi_{s_n+j},\tau}(\mathbf{r}_{\xi(p)}^j)\}_{j=1}^{k},\ \tau(\mathsf{id}^*))$, where $s_i = (i-1)k$
- $\mathbf{c}_3 = \mathsf{COM}(\{P_{\pi_j,\phi_j,\tau}(\mathbf{x}_{\xi(1)}^j + \mathbf{r}_{\xi(1)}^j)\}_{j=1}^{k},\ \ldots\ ,\ \{P_{\pi_{s_n+j},\phi_{s_n+j},\tau}(\mathbf{x}_{\xi(p)}^j + \mathbf{r}_{\xi(p)}^j)\}_{j=1}^{k},$ $\tau(\mathsf{id}^* + \mathbf{r}_{\mathsf{id}^*}))$, where $s_i = (i-1)k$

Challenge: The verifier generates a challenge $CH \hookleftarrow \{1,2,3\}$ uniformly at random.

Response: The response is computed by the prover depending on the outcome of CH. We differentiate 3 cases:

- $\mathbf{CH} = \mathbf{1}$: The response is composed as follows. For $i \in [p]$ compute $\{\mathbf{b}_i^j = P_{\pi_{s_i+j},\phi_{s_i+j},\tau}(\mathbf{r}_{\xi(i)}^j)\}_{j=1}^k$, $\{\mathbf{w}_i^j = P_{\pi_{s_i+j},\phi_{s_i+j},\tau}(\mathbf{x}_{\xi(i)}^j)\}_{j=1}^k$ for $s_i = (i-1)k$. Furthermore, determine $\mathbf{p}_{\mathsf{id}^*} = \tau(\mathsf{id}^*)$ and $\mathbf{b}_{\mathsf{id}^*} = \tau(\mathbf{r}_{\mathsf{id}^*})$. Output

$$RSP_1 = \{\ \{\mathbf{b}_1^j,\ \mathbf{w}_1^j\}_{j=1}^k,\ \dots\ ,\{\mathbf{b}_p^j,\ \mathbf{w}_p^j\}_{j=1}^k,\ \mathbf{p}_{\mathsf{id}^*},\ \mathbf{b}_{\mathsf{id}^*}\}\,.$$

- $\mathbf{CH} = \mathbf{2}$: For $i \in [n]$ compute $\{\mathbf{v}_i^j = \mathbf{x}_i^j + \mathbf{r}_i^j\}_{j=1}^k$. Determine and $\mathbf{v}_{\mathsf{id}^*} = \mathsf{id}^* + \mathbf{r}_{\mathsf{id}}^*$. The response is then given by

$$RSP_2 = \{\tau,\ \{\phi_j\}_{j=1}^{p\cdot k},\ \{\pi_j\}_{j=1}^{p\cdot k},\ \xi,\ \{\mathbf{v}_1^j\}_{j=1}^k,\ \dots\ ,\ \{\mathbf{v}_p^j\}_{j=1}^k,\ \mathbf{v}_{\mathsf{id}^*}\}\,.$$

- $\mathbf{CH} = \mathbf{3}$: The prover needs only to output the response

$$RSP_3 = \{\tau,\ \{\phi_j\}_{j=1}^{p\cdot k},\ \{\pi_j\}_{j=1}^{p\cdot k},\ \xi,\ \{\mathbf{r}_1^j\}_{j=1}^k,\ \dots\ ,\ \{\mathbf{r}_p^j\}_{j=1}^k,\ \mathbf{r}_{\mathsf{id}^*}\}\,.$$

Verification The verifier requires always to check only 2 out of 3 commitments, as otherwise the availability of responses to all 3 commitments allows to deduce the witness.

- $\mathbf{CH} = \mathbf{1}$:Given RSP_1, check that $\mathbf{p}_{\mathsf{id}^*} \in \mathsf{B}_{2\ell}$ and $\mathbf{w}_i^j$ is valid with respect to $\mathbf{p}_{\mathsf{id}^*}$ for at least t set of vectors and all $j \in [k]$. Furthermore, verify that
  - $\mathbf{c}_2 = \mathsf{COM}(\{\mathbf{b}_1^j\}_{j=1}^k,\ \dots\ ,\{\mathbf{b}_p^j\}_{j=1}^k,\ \mathbf{b}_{\mathsf{id}^*})$
  - $\mathbf{c}_3 = \mathsf{COM}(\{\mathbf{w}_1^j + \mathbf{b}_1^j\}_{j=1}^k,\ \dots\ ,\{\mathbf{w}_p^j + \mathbf{b}_p^j\}_{j=1}^k,\ \mathbf{p}_{\mathsf{id}^*} + \mathbf{b}_{\mathsf{id}^*})$
- $\mathbf{CH} = \mathbf{2}$: On input RSP_2 the verifier takes ξ, computes $\bar{\mathbf{v}}_i^j = P_{\pi_{s_i+j},\phi_{s_i+j},\tau}(\mathbf{v}_{\xi(i)}^j)$ for $s_i = (i-1)k$ and verifies that
  - $\mathbf{c}_1 = \mathsf{COM}([\ \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j \mathbf{v}_1^j) - \mathbf{u}_1, \dots, \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j \mathbf{v}_p^j) - \mathbf{u}_p],$ $\tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ ,\ \xi)$
  - $\mathbf{c}_3 = \mathsf{COM}(\{\bar{\mathbf{v}}_1^j\}_{j=1}^k,\ \dots\ ,\ \{\bar{\mathbf{v}}_p^j\}_{j=1}^k,\ \tau(\mathbf{v}_{\mathsf{id}^*}))$
- $\mathbf{CH} = \mathbf{3}$: On input RSP_3 the verifier takes ξ, computes $\bar{\mathbf{r}}_i^j = P_{\pi_{s_i+j},\phi_{s_i+j},\tau}(\mathbf{r}_{\xi(i)}^j)$ for $s_i = (i-1)k$ and verifies that
  - $\mathbf{c}_1 = \mathsf{COM}([\ \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j \mathbf{r}_1^j),\ \dots\ ,\mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j \mathbf{r}_p^j)],$ $\tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi)$
  - $\mathbf{c}_2 = \mathsf{COM}(\{\bar{\mathbf{r}}_1^j\}_{j=1}^k,\ \dots\ ,\ \{\bar{\mathbf{r}}_p^j\}_{j=1}^k,\ \tau(\mathbf{r}_{\mathsf{id}^*}))$

Reducing the Communication Costs. We observe that the communication costs can be reduced in terms of permutations. In the sZKAoK the prover has to provide the set of permutations, whenever the commitment $\mathbf{c}_1$ is being verified. As opposed to revealing the permutations themselves or including them, when computing the commitment $\mathbf{c}_1$, we reveal the underlying randomness instead, i.e. we generate the set of permutations by use of a random function $F(\cdot)$ such as a pseudo random number generator taking as input a uniform random string $\mu \in \{0,1\}^*$ of sufficient entropy. All permutations can thus deterministically be deduced from μ. As a result, we can replace all permutations in $\mathbf{c}_1$ by μ. The prover requires only to reveal μ, if required.

$$F : \mathbb{N}^3 \times \{0,1\}^* \to \mathsf{S}_{2\ell} \times \mathsf{S}^*_{3m} \times \mathsf{S}^*_{3m} \times \mathsf{S}_n \times \mathsf{S}_{3d}$$
$$F(n, k, \bar{k}, \mu) = (\tau, \{\phi_j\}_{j=1}^{p\cdot k}, \{\pi_j\}_{j=1}^{p\cdot k}\}, \xi)$$

6.1 Features of Statistical Zero-Knowledge Arguments of Knowledge

In this section, we show how to extend the sZKAoK from [26] in order to cover the new functionality. In particular, we prove that our construction still maintains the desired properties of statistical zero-knowledge (even for the position of the real credentials) and special soundness. Thus let COM be a statistically hiding and computationally binding commitment scheme. In the next sections we will step by step prove that our construction is a sZKAoK for the language

$$\begin{aligned} L(n,k,\ell,m,\gamma,p,t) = \{&\mathsf{public} := \{\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n\times m}; \mathbf{u}_1, \ldots, \mathbf{u}_p\}, \\ &\mathsf{witness} := \{\mathsf{id} \in \{0,1\}^\ell, \mathbf{z}_{j_1}, \ldots, \mathbf{z}_{j_t} \in \mathbb{Z}^{2m}\} : \\ &(\|\mathbf{z}_i\|_\infty \le \gamma \ \wedge \ [\ \mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \mathsf{id}_j \mathbf{A}_j\ ] \cdot \mathbf{z}_i = \mathbf{u}_i \bmod q \\ &\text{ for } i \in \{j_1, \ldots, j_t\})\}. \end{aligned}$$

We mainly follow the proof techniques of Stern-type protocols.

Theorem 4. *For a a statistically hiding and computationally binding commitment scheme* COM*, the protocol given in Section 6 is a* sZKAoK *for the language* $L(n,k,d,\ell,m,\gamma,\beta,t)$*, where the execution of each round has perfect completeness, soundness error* $2/3$*, argument of knowledge property and communication cost* $O(p\ell m \log\beta)\log q$.

The proof of this theorem is given within the following subsections.

6.2 Communication Cost

When using the commitment scheme due to Kawachi et al. [19] the output of COM is $n \log q$ bits. Thus the prover sends 3 commitments at the start of the interaction amounting to $3n \log q$ bits. The challenger subsequently responds with a 2 bit challenge $CH \in \{1, 2, 3\}$. Let p denote the number of attributes and $k = \lfloor \log \gamma \rfloor + 1$ be the number of basis elements for the interval $[0, \gamma]$, then the responses composed by the prover include elements from

Masking Terms

- pk vectors from $\mathbb{Z}_q^{(2\ell+2)3m}$
- 1 vector from $\mathbb{Z}_q^{2\ell}$

Permutations

- 1 permutation for n elements
- 1 permutation for 2ℓ elements
- $2pk$ permutations for $(2\ell + 2)3m$ elements.

Thus, the overall communication cost is upper bounded by $O(p\ell m \log \beta) \log q$ bits.

6.3 Completeness

The completeness requirement ensures that an honest prover P, that possesses valid credentials $(\mathsf{id}^*, \mathbf{z}_{j_1}, \ldots, \mathbf{z}_{j_t})$ and follows the protocol for a policy $\psi = \{t \text{ out of } B \subseteq \mathsf{Attributes}\}$, should be able to generate a proof for given public input $(\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}_1, \ldots, \mathbf{u}_p)$ such that it successfully satisfies the verification checks by V. In fact, he prepares $\{\tilde{\mathbf{z}}_i^j \in \mathsf{B}_{3m}\}_{j=1}^k$ for $i \in \{j_1, \ldots, j_t\}$ such that

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot f_{\mathsf{id}^*}(\tilde{\mathbf{z}}_i^j) \right) = \bar{\mathbf{A}} \cdot f_{\mathsf{id}}(\mathbf{z}_i) \equiv \mathbf{u}_i \, .$$

The same is accomplished with fake credentials $\{\mathbf{d}_i^j \in \mathbb{Z}_q^{(2+2\ell)3m}\}_{j=1}^{p-t}$ for $i \in [p] \backslash \{j_1, \ldots, j_t\}$ such that

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \mathbf{d}_i^j \right) \equiv \mathbf{u}_i \, .$$

We now prove that P correctly computes the responses passing the verification checks for all $CH \in \{1, 2, 3\}$.

- $\mathbf{CH = 1}$ Since $\mathsf{id}^* \in \mathsf{B}_{2\ell}$ and the set $\mathsf{B}_{2\ell}$ is invariant under permutations, we have $\mathbf{p}_{\mathsf{id}^*} \in \mathsf{B}_{2\ell}$ and $\mathbf{w}_i^j$ is valid with respect to $\mathbf{p}_{\mathsf{id}^*}$ for at least t set of vectors and all $j \in [k]$, i.e. $\mathbf{w}_i^j \in \{-1, 0, 1\}^{(2+2\ell)3m}$ and has zero blocks at the zero-positions of $\mathbf{p}_{\mathsf{id}^*}$.

- **CH = 2** The honest prover should be able to generate $\mathbf{w}_i^j$ and $\mathbf{r}_i^j$ such that the following expressions are true

$$\begin{gathered}\mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{w}_1^j) - \mathbf{u}_1 = \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{r}_1^j) \\ \cdots \\ \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{w}_p^j) - \mathbf{u}_N = \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{r}_N^j)\,.\end{gathered}$$

6.4 Statistical Zero-Knowledge Property

The zero-knowledge property is shown as follows. In particular, we construct a simulator $\mathcal{S}$ interacting with the verifier for some given public input. The simulator outputs with probability $2/3 - \mathsf{negl}(n)$ a simulated transcript that is statistically close to an honestly generated transcript by the prover in the real interaction. The simulator predicts the challenge to take values $CH_1 = 2, 3$, $CH_2 = 1, 3$ or $CH_3 = 1, 2$.

- **Case A** ($CH = 2, 3$)
  - Generate $\mathbf{x}_i^j \hookleftarrow \mathbb{Z}_q^{(2+2\ell)3m}$ for $i \in [p], j \in [k]$ such that

$$\mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{x}_i^j) = \mathbf{u}_i \bmod q$$

  - Generate $\mathsf{id} \in \mathbb{Z}_q^{2\ell}$
  - Sample uniform random $\mathbf{r}_i^j \hookleftarrow \mathbb{Z}_q^{(2+2\ell)3m}$ for $i \in [p], j \in [k]$ and $\mathbf{r}_{\mathsf{id}} \hookleftarrow \mathbb{Z}_q^{2\ell}$
  - Sample permutations $\tau \hookleftarrow \mathsf{S}_{2\ell}, \{\phi_j \hookleftarrow \mathsf{S}_{3m}\}_{j=1}^{p\cdot k}, \{\pi_j \hookleftarrow \mathsf{S}_{3m}\}_{j=1}^{p\cdot k}$ (for each attribute k permutations), $\xi \hookleftarrow \mathsf{S}_n$

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

- $\mathbf{c}_1' = \mathsf{COM}([\,\mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{r}_1^j), \ldots, \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^{k} \gamma_j \mathbf{r}_p^j)],\ \tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi)$

- $\mathbf{c}_2' = \mathsf{COM}(\{P_{\pi_j,\phi_j,\tau}(\mathbf{r}_{\xi(1)}^j)\}_{j=1}^{k},\ \ldots\ ,\ \{P_{\pi_{s_n+j},\phi_{s_n+j},\tau}(\mathbf{r}_{\xi(p)}^j)\}_{j=1}^{k},\ \tau(\mathsf{id}))$, where $s_i = (i-1)k$

- $\mathbf{c}_3' = \mathsf{COM}(\{P_{\pi_j,\phi_j,\tau}(\mathbf{x}_{\xi(1)}^j + \mathbf{r}_{\xi(1)}^j)\}_{j=1}^{k},\ \ldots\ ,\ \{P_{\pi_{s_n+j},\phi_{s_n+j},\tau}(\mathbf{x}_{\xi(p)}^j + \mathbf{r}_{\xi(p)}^j)\}_{j=1}^{k},$ $\tau(\mathsf{id} + \mathbf{r}_{\mathsf{id}}))$, where $s_i = (i-1)k$

If the verifier selects
$CH = 1$, $\mathcal{S}$ outputs $\perp$ and aborts.
$CH = 2$, $\mathcal{S}$ outputs

$$RSP_2 = \{\{\mathbf{x}_1^j + \mathbf{r}_1^j\}_{j=1}^{k}, \ldots, \{\mathbf{x}_N^j + \mathbf{r}_N^j\}_{j=1}^{k}, \mathsf{id} + \mathbf{r}_{\mathsf{id}},\ \tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi\}$$

$CH = 3$, $\mathcal{S}$ outputs

$$RSP_3 = \{\{\mathbf{r}_1^j\}_{j=1}^k, \ldots, \{\mathbf{r}_N^j\}_{j=1}^k, \mathbf{r}_{\mathsf{id}},\ \tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi\}$$

- **Case B** ($CH = 1, 3$)
  - Generate $\mathsf{id} \in \mathsf{B}_{2\ell}$
  - Sample $\mathbf{x}_i^j \hookleftarrow \mathsf{B}_{(2+2\ell)3m}$ for $i \in \{i_1, \ldots, i_t\}, j \in [k]$, that are valid with respect to id, and $\mathbf{x}_i^j \hookleftarrow \mathbb{Z}_q^{(2+2\ell)3m}$ for $i \in [p]\backslash\{i_1, \ldots, i_t\}, j \in [k]$.
  - Sample uniform random $\mathbf{r}_i^j \hookleftarrow \mathbb{Z}_q^{(2+2\ell)3m}$ for $i \in [p], j \in [k]$ and $\mathbf{r}_{\mathsf{id}} \hookleftarrow \mathbb{Z}_q^{2\ell}$
  - Sample permutations $\tau \hookleftarrow \mathsf{S}_{2\ell}, \{\phi_j \hookleftarrow \mathsf{S}_{3m}\}_{j=1}^{p\cdot k}, \{\pi_j \hookleftarrow \mathsf{S}_{3m}\}_{j=1}^{p\cdot k}$ (for each attribute k permutations), $\xi \hookleftarrow \mathsf{S}_n$

The simulator $\mathcal{S}$ generates commitments $CMT = (\mathbf{c}_1', \mathbf{c}_2', \mathbf{c}_3')$ as in the previous case and sends them to the verifier V.

If the verifier selects
$CH = 1$, $\mathcal{S}$ outputs

$$\begin{aligned} RSP_2 = \{&\{P_{\pi_j,\phi_j,\tau}(\mathbf{x}_{\xi(1)}^j)\}_{j=1}^k,\ \ldots\ ,\ \{P_{\pi_{s_N+j},\phi_{s_N+j},\tau}(\mathbf{r}_{\xi(p)}^j)\}_{j=1}^k, \\ &\{P_{\pi_j,\phi_j,\tau}(\mathbf{r}_{\xi(1)}^j)\}_{j=1}^k,\ \ldots\ ,\ \{P_{\pi_{s_p+j},\phi_{s_p+j},\tau}(\mathbf{r}_{\xi(p)}^j)\}_{j=1}^k,\ \tau(\mathsf{id}), \tau(\mathbf{r}_{\mathsf{id}})\}\,. \end{aligned}$$

$CH = 2$, $\mathcal{S}$ outputs $\perp$ and aborts.
$CH = 3$, $\mathcal{S}$ outputs

$$RSP_3 = \{\{\mathbf{r}_1^j\}_{j=1}^k, \ldots, \{\mathbf{r}_p^j\}_{j=1}^k, \mathbf{r}_{\mathsf{id}},\ \tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi\}\,.$$

- **Case C** ($CH = 2, 3$) The main difference to the previous case is the way $\mathbf{c}_1'$ is generated. More precisely, $\mathcal{S}$ computes
  $\mathbf{c}_1' = \mathsf{COM}([\ \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j(\mathbf{x}_1^j + \mathbf{r}_1^j)) - \mathbf{u}_1, \ldots, \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j(\mathbf{x}_p^j + \mathbf{r}_p^j)) - \mathbf{u}_p],\ \tau,\ \{\pi_j\}_{j=1}^{p\cdot k},\ \{\phi_j\}_{j=1}^{p\cdot k},\ \xi)\,.$

If the verifier selects
$CH = 1$, $\mathcal{S}$ outputs RSP following **Case B**, $CH = 1$.
$CH = 2$, $\mathcal{S}$ outputs RSP following **Case A**, $CH = 2$.
$CH = 3$, $\mathcal{S}$ outputs $\perp$ and aborts.

Based on the statistically hiding property of $\mathsf{COM}(\cdot)$ the distribution of the commitments and challenges are statistically close to those in a real interaction. Thus, the probability that $\mathcal{S}$ outputs $\perp$ and aborts is $1/3 - \mathsf{negl}(n)$, otherwise he outputs valid transcripts, that are distributed statistically close to those in a real interaction. Thus, the so constructed simulator can impersonate an honest prover with probability $2/3 + \mathsf{negl}(n)$.

6.5 Argument of Knowledge

In the following section we will show that the protocol satisfies the *special soundness* property for the language $L(n, k, \ell, m, \gamma, p, t)$, i.e. if there exists a prover who is able to simultaneously provide valid responses to all 3 challenges (i.e. $CH = 1, 2, 3$) satisfying the same commitment CMT, then there exists a polynomial-time witness extractor $\mathcal{K}$ that outputs $(\mathsf{id}, \mathbf{z}_{j_1}, \ldots, \mathbf{z}_{j_t}, \mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_\ell, \mathbf{u}_1, \ldots, \mathbf{u}_p) \in L(n, k, \ell, m, \gamma, p, t)$.

The extractor $\mathcal{K}$ obtains $\{RSP_j\}_{j=1}^3$, that satisfy CMT. In particular, he obtains:

- $RSP_1 = \{ \ \{\mathbf{b}_1^j, \ \mathbf{w}_1^j\}_{j=1}^k, \ \ldots \ , \{\mathbf{b}_p^j, \ \mathbf{w}_p^j\}_{j=1}^k, \ \mathbf{p}_{\mathsf{id}^*}, \ \mathbf{b}_{\mathsf{id}^*} \}$.
- $RSP_2 = \{\tau, \ \{\phi_j\}_{j=1}^{p\cdot k}, \ \{\pi_j\}_{j=1}^{p\cdot k}, \ \xi, \ \{\mathbf{v}_1^j\}_{j=1}^k, \ \ldots \ , \ \{\mathbf{v}_p^j\}_{j=1}^k, \ \mathbf{v}_{\mathsf{id}^*} \}$.
- $RSP_3 = \{\tau, \ \{\phi_j\}_{j=1}^{p\cdot k}, \ \{\pi_j\}_{j=1}^{p\cdot k}, \ \xi, \ \{\mathbf{r}_1^j\}_{j=1}^k, \ \ldots \ , \ \{\mathbf{r}_p^j\}_{j=1}^k, \ \mathbf{r}_{\mathsf{id}^*} \}$.

Since $\mathbf{w}_{i'}^j = \mathbf{w}_{\xi(i)}^j$ for some i, we can reorder $\mathbf{v}_{i'}^j, \mathbf{b}_{i'}^j, \ \mathbf{w}_{i'}^j$ to $\mathbf{v}_i^j, \mathbf{b}_i^j, \ \mathbf{w}_i^j$ by use of the inverse ξ^{-1}. Subsequently, we note that $\{\mathbf{w}_{\xi(i)}^j\}_{j=1}^k$ are valid with respect to $\mathbf{p}_{\mathsf{id}^*}$ for at least t set of vectors. It holds that $\mathbf{p}_{\mathsf{id}^*} + \mathbf{r}_{\mathsf{id}^*} = \tau(\mathbf{v}_{\mathsf{id}^*})$. Thus, we apply the inverse of P to each set of vectors, i.e. $\{\mathbf{x}_i^j = P^{-1}_{\pi_{s_i+j}, \phi_{s_i+j}, \tau}(\mathbf{w}_i^j)\}_{j=1}^k$ for $s_i = (i-1)k$ and $i \in [p]$. We deduce $\mathsf{id}^* = (\mathsf{id}_1, \ldots, \mathsf{id}_\ell, \ldots, \mathsf{id}_{2\ell}) = \tau^{-1}(\mathbf{p}_{\mathsf{id}^*})$ and obtain $\mathsf{id} = (\mathsf{id}_1, \ldots, \mathsf{id}_\ell)$.
Furthermore, we obtain $\{\mathbf{x}_i^j\}_{j=1}^k$, where at least t sets have elements in $\mathsf{B}_{(2+2\ell)3m}$ and are valid with respect to id^*, i.e. $\mathbf{x}_i^j = (\mathbf{x}_{i,1}^j, \mathbf{x}_{i,2}^j, \mathsf{id}_1 \cdot \mathbf{x}_{i,2}^j, \ldots, \mathsf{id}_{2\ell} \cdot \mathbf{x}_{i,2}^j)$, since $\mathbf{w}_i^j$ are valid with respect to $\mathbf{p}_{\mathsf{id}^*}$. Furthermore, it holds that $\mathbf{v}_i^j = \mathbf{x}_i^j + \mathbf{r}_i^j$ with $\mathbf{x}_i = \sum_{j=1}^k \gamma_j \mathbf{x}_i^j$ and $\|\mathbf{x}_i\| \leq \gamma$ for t set of vectors such that

$$\mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j \mathbf{v}_i^j) - \mathbf{u}_i = \mathbf{A}_{\mathsf{ext}} \cdot (\sum_{j=1}^k \gamma_j \mathbf{r}_i^j).$$

This implies $\mathbf{A}_{\mathsf{ext}} \cdot \mathbf{x}_i = \mathbf{u}_i$ such that $\mathbf{z}_i = f_{\mathsf{id}^*}^{-1}(\mathbf{x}_i)$ is a valid signature for $(\mathbf{A}_{\mathsf{id}}, \mathbf{u}_i)$.

7 Attribute-based Signature Scheme with Expressive Policies

In this section we present our attribute-based signature scheme for expressive policies. This is accomplished by use of our threshold ABS scheme introduced in Section 5. We furthermore require a mechanism that aggregates the signatures of a certain user id on different attributes. Thus, we first develop an attribute aggregation scheme and combine it with the threshold ABS scheme in order to allow for expressive policies.

An attribute aggregation scheme allows a certain signer possessing signatures on different attributes $\mathbf{u}_j \in B \subseteq \mathsf{Attributes}$ to generate a single aggregate signature on all those attributes, i.e. for the policy $\psi_B = \bigwedge_{\mathbf{u}_j \in B} U_j$, where $U_j \in \{0, 1\}$.

This allows to hide the number of used attributes within the zero-knowledge argument. We omit a description of the key generation step as the aggregation scheme is built upon the ABS scheme from Section 5.

Attribute Aggregation Scheme for $\bigwedge$-Policies

- $\mathsf{AggAttribute}(\mathsf{sk}_{\mathsf{id}}, B \subseteq \mathsf{Attributes})$: On input a set of valid attributes and set of signature attribute pairs $\mathsf{sk}_{\mathsf{id}}$, the user checks that he possesses for all attributes $\mathbf{u} \in B$ a valid signature. Subsequently, he generates a signature
$$\mathbf{z}_B = \sum_{\substack{(\mathbf{z}_j,\mathbf{u}_j)\in \mathsf{sk}_{\mathsf{id}} \wedge \\ \mathbf{u}_j \in B}} \mathbf{z}_j \text{ for } \psi_B = \bigwedge_{\mathbf{u}_j \in B} U_j,$$
where ψ_B denotes the associated conjunction policy. The value of the j-th decision variable is $U_j = 1$ if the user possesses a valid signature for $\mathbf{u}_j \in B$, else $U_j = 0$. If the user does not possess a valid signature on an attribute in B, he outputs $\perp$.
- $\mathsf{AggVerify}(\mathbf{z}_B, B \subseteq \mathsf{Attributes})$: On input the aggregate signature $\mathbf{z}_B$ and the related set of attributes B, check that
$$\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z}_B = \sum_{\mathbf{u}_j \in B} \mathbf{u}_j, \ \|\mathbf{z}_B\| \leq \sqrt{|B|} \cdot \gamma\,.$$
If satisfied output 1, else 0.

The security of the scheme is based on the fact, that it is hard for a fixed user to generate small vectors that map to one or more attributes. We therefore give a security model capturing this idea. We note that different users cannot collude in order to generate valid signatures on attributes due to differing public keys $\mathbf{A}_{\mathsf{id}}$ and signature parts related to $\sum_{i=1}^{\ell} \mathsf{id}_i \cdot \mathbf{A}_i$ are not zero. Thus, we consider in the following experiment adversaries against a fixed identity. To this end, we allow the adversary to receive from the attribute authority signatures on attributes of choice. Eventually, he outputs an aggregate signature on a policy that contains at least one attribute, for which he did not receive any signature.

Experiment: $\mathsf{Exp}_{\mathcal{A}}^{Agg}(n, \mathsf{Attributes})$
- $(\mathbf{T}, \mathbf{A}) \leftarrow \mathsf{KeyGen}(1^n)$
- $(\mathbf{z}_B^*, B^* \subseteq \mathsf{Attributes}, \mathsf{St}) \leftarrow \mathcal{A}^{\mathsf{Sign}(\mathbf{T},\cdot),\mathsf{AggAttribute}(\cdot,\cdot),\mathsf{AggVerify}(\cdot,\cdot)}(\mathbf{A}, \mathsf{Attributes})$
- If all $\mathbf{u}_i \in B^*$ were queried to Sign, return 0.
- Return 1 if $\mathsf{AggVerify}(\mathbf{z}_B^*, B^* \subseteq \mathsf{Attributes}) = 1$, else 0.

Theorem 5. *If there exists a* PPT *adversary that wins the game* $\mathsf{Exp}_{\mathcal{A}}^{Agg}$ *for a subset of attributes* $B \subseteq \mathsf{Attributes}$*, then there exists a* PPT *algorithm* $\mathcal{M}$ *that solves* $SIS_{n,m,\delta}$ *for* $\delta \leq 2\sqrt{|B|} \cdot \gamma$ *and* $|B| \leq |\mathsf{Attributes}|$.

Proof. Let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ denote the problem instance to algorithm $\mathcal{M}$. The goal of $\mathcal{M}^{\mathcal{A}}$ is to output a solution to the SIS problem $\mathbf{A} \cdot \mathbf{x} \equiv \mathbf{0} \bmod q$, where $\|\mathbf{z}\| \leq \delta$. Let $p := |\mathsf{Attributes}|$ be the number of attributes.

Setup. The algorithm $\mathcal{M}$ maintains a list $Q[\cdot]$ which is filled at the beginning with $Q[i] = (\mathbf{z}_i \hookleftarrow D_{\mathbb{Z}^{2m},s}, \mathbf{u}_i = \mathbf{A}\mathbf{z}_i \bmod q, 0)$ for $i \in [p]$. The list of attributes is set to $\mathsf{Attributes} = \{\mathbf{A}\mathbf{z}_1, \ldots, \mathbf{A}\mathbf{z}_p\}$

Signing Queries. If the signing oracle is queried on some $\mathbf{u}_j \in \mathsf{Attributes}$, the algorithm $\mathcal{M}$ looks up in the Q-List for an entry $(\mathbf{z}, \mathbf{u}, *)$ such that $\mathbf{u}_j = \mathbf{u}$, outputs $\mathbf{z}$ and sets $(\mathbf{z}, \mathbf{u}, 1)$, else $\mathcal{M}$ aborts. He also aborts, if all entries have been set to $(\mathbf{z}_i, \mathbf{u}_i, 1)$ for $i \in [p]$, because the adversary has then obtained signatures for all attributes.

Queries to $\mathsf{AggAttribute}(\cdot,\cdot)$. If the oracle $\mathsf{AggAttribute}(\cdot,\cdot)$ has been queried on input a subset $B \subseteq \mathsf{Attributes}$ and a set of tuples $\{(\mathbf{x}_j, \mathbf{u}_j)\}_{j=1}^{|B|}$, the algorithm $\mathcal{M}$ computes $\mathbf{x} = \sum_{j=1}^{|B|} \mathbf{x}_j$ and outputs $\mathbf{x}$.

Queries to $\mathsf{AggVerify}(\cdot,\cdot)$. If the oracle $\mathsf{AggVerify}(\cdot,\cdot)$ has been queried on input a subset $B \subseteq \mathsf{Attributes}$ and a vector $\mathbf{x}$, the algorithm $\mathcal{M}$ checks that indeed $B \subseteq \mathsf{Attributes}$, $\|\mathbf{x}\| \leq \sqrt{|B|}\gamma$ and $\mathbf{A}\mathbf{x} = \sum_{j=1}^{|B|} \mathbf{u}_j$, where $\mathbf{u}_j \in B$. If satisfied, he outputs 1, else 0.

Eventually, the adversary $\mathcal{A}$ outputs a valid tuple $(\mathbf{z}^*, B^* \subseteq \mathsf{Attributes})$ such that $\mathsf{AggVerify}(\mathbf{z}^*_{B^*}, B^*) = 1$ and B^* contains an attribute, for which $\mathcal{A}$ has not queried the signing oracle. Let $\mathbf{u}^*$ denote this attribute. Then, $\mathcal{M}$ computes $\mathbf{x} = \mathbf{z}^* - \sum_{\mathbf{u}_j \in B} \mathbf{z}_j$, which is due to the high min-entropy of a discrete Gaussian not equal to zero and hence represents a valid solution to SIS with $\mathbf{A} \cdot \mathbf{x} = \mathbf{0} \bmod q$ and $\|\mathbf{x}\| \leq 2\sqrt{|B|}\gamma$. □

This aggregation mechanism can be seen as a further property of Boyen's signature scheme or its modified variant. Furthermore, it allows to reduce the communication cost, when proving to the verifier that the user possesses valid signatures for attributes in $\bigwedge$-policies, since the aggregate signature represents just a single vector. In addition to that, it essentially hides the number of aggregated signatures, since for the sZKAoK the upper bound can be set such that all aggregate signatures underlying $\bigwedge$-policies satisfy this bound. Based on these tools we can construct ABS schemes for arbitrary $(\wedge, \vee)$-policies. In the following section we present our construction that is built upon the threshold ABS scheme instantiated with the modified variant of Boyen's signature scheme (Section 3.1) and its attribute aggregation property.

7.1 Construction

The generic ABS scheme for expressive policies essentially represents an instance of an **1-out-of-l**-threshold ABS scheme introduced in Section 5, i.e. we expand an arbitrary $(\wedge, \vee)$-policy ψ into its disjunctive normal form (DNF) $\psi = C_1 \vee \ldots \vee C_l$ for $\bigwedge$-policies C_i and show via the aggregation scheme that a user satisfies ψ, if he possesses a valid aggregate signature for any of the C_i.

In fact, we do not even need to introduce span programs as required in classical ABS schemes. Consequently, this relieves the signer from standard techniques such as secret sharing schemes.

$\mathsf{Setup}_{\mathsf{E}}(1^{\lambda})$: The public parameters are set to n and q and the discrete Gaussian parameter s.

$\mathsf{AASetup}_{\mathsf{E}}(\mathsf{pp}, \mathsf{aid})$: The attribute authority generates $\ell + 2$ public matrices $\mathsf{vk}_{\mathsf{AA}} := \{\mathbf{A}, \mathbf{A}_0, \ldots, \mathbf{A}_{\ell} \in \mathbb{Z}_q^{n \times m}\}$ as the public key and the associated trapdoor $\mathsf{sk}_{\mathsf{AA}} := \mathbf{T}$ according to the modified Boyen's signature scheme in Section 3.1, where 2^{ℓ} denotes the number of users. Furthermore he generates a set of attributes

$$\mathsf{Attributes} = \{\mathbf{u}_1, \ldots, \mathbf{u}_p\},$$

where $\mathbf{u}_i \in \mathbb{Z}_q^n$ is uniform random. Each attribute at_i is associated to a uniform random element $\mathbf{u}_i$, for instance via a public list of tuples $Q[i] = (\mathsf{at}_i, \mathbf{u}_i)$.

$\mathsf{AttKeyGen}_{\mathsf{E}}(\mathsf{pp}, \mathsf{id}, B \subseteq \mathsf{Attributes}, \mathsf{sk}_{\mathsf{AA}})$: A certain user represented as a bit string $\mathsf{id} \in \{0,1\}^{\ell}$ is assigned a set of attributes

$$\mathsf{sk}_{\mathsf{id}} = \{(\mathbf{z}_{\mathsf{id},j_1}, \mathbf{u}_{j_1}), \ldots, (\mathbf{z}_{\mathsf{id},j_k}, \mathbf{u}_{j_k})\}$$

by the attribute authority using its secret key $\mathsf{sk}_{\mathsf{AA}}$ such that

$$\mathbf{A}_{\mathsf{id}} \cdot \mathbf{z}_{\mathsf{id},j_1} = \mathbf{u}_{j_1} \bmod q, \;\; \|\mathbf{z}_{\mathsf{id},j_1}\| \leq \gamma$$

is satisfied for $\mathbf{u}_j \in B$.

$\mathsf{Sign}_{\mathsf{E}}(\mathsf{pp}, m, \psi, \mathsf{sk}_{\mathsf{id}})$: On input the message m, a policy ψ and the secret key, the signer id expands ψ into its DNF

$$\psi = C_1 \vee \ldots \vee C_l,$$

where ψ_{C_i} denotes a $\bigwedge$-policy for $i \in [l]$. We have

$$\psi = 1 \iff \exists\ C_i \text{ s.th. } \psi_{C_i} = 1\,.$$

Each C_l is associated to a set of attributes $B_l \subseteq \mathsf{Attributes}$. Further, suppose that the user possesses valid credentials for the policy $C_j = \bigwedge_{\mathbf{u}_h \in B_j} U_h$ and the associated set of attributes B_j. The user generates a proof of knowledge

$$\Pi = \mathsf{SPoK}(\mathsf{public} := \{m, \mathsf{Attributes}, \mathbf{A}\},\ \mathsf{witness} := \{\mathsf{sk}_{id}\} : \\ \exists\ B \subseteq \mathsf{sk}_{\mathsf{id}} \wedge C^* \text{ s.th. } \psi(B) = \psi_{C^*}(B) = 1)$$

Output signature $\Sigma = (m, \Pi)$.

$\mathsf{Verify}_{\mathsf{E}}(\mathsf{pp}, \Sigma, \psi, \mathsf{vk}_{\mathsf{AA}}, \mathsf{Attributes})$: On input the policy, the list of attributes, the public verification matrix $\mathbf{A}$ and an ABS signature, which is parsed as $\Sigma = (m, \Pi)$, the verifier returns 1 if $\Pi = \mathsf{SPoK}$ is a valid proof, otherwise he outputs 0.

7.2 Informal Description

We briefly describe, how to anonymously sign a message, in case the user owns signatures on attributes satisfying the policy. The key generation step is exactly the same as for the threshold ABS scheme from Section 5. However, we need to modify the signing step in order to allow for expressive $(\wedge, \vee)$-policies ψ.

1. Expand ψ into its DNF form $\psi = C_1 \vee \ldots \vee C_l$, where
$$C_j = \bigwedge_{\mathbf{u}_h \in B_j} U_h$$
for the associated set of attributes $B_j \subseteq \mathsf{Attributes}$. The boolean variables $U_h \in \{0,1\}$ are equal to 1, if the user possesses valid signatures on the corresponding attributes, else 0. Suppose that the user satisfies ψ for one disjunctive term C^*. Let B^* denote the associated set of attributes.
2. Determine
$$d_{max} = \max_{i \in [l]} |B_i|,$$
which denotes the maximum number of attributes per set. Then, the user defines the upper bound on the size of a valid aggregate signature such that $\|\mathbf{z}_{C_i}\| \leq \sqrt{d_{max}} \cdot \gamma$ for all $i \in [l]$, where γ denotes an upper bound on the size of a single signature as before. This follows from Lemma 3 for sums of discrete Gaussians.
3. The user generates real credentials and fake credentials as for the threshold ABS scheme. In fact, we generate an aggregate signature satisfying C^* and fake credentials for $C_i \neq C^*$ and apply an instance of the **1**-out-of-**l** threshold ABS scheme.

   - **Real Credential**: For C^* and B^* the user retrieves the corresponding signatures from $\mathsf{sk}_{\mathsf{id}}$ and generates
$$\mathbf{z}_{B^*} = \sum_{\mathbf{u}_j \in B^*} \mathbf{z}_j \,.$$
Let $k = \lfloor \log(\gamma \sqrt{d_{max}}) \rfloor + 1$ be the parameter as for the threshold ABS scheme. By use of the Extension-Decomposition technique applied on $\mathbf{z}_{B^*}$ we obtain one set of vectors $\{\tilde{\mathbf{z}}^j_{B^*} \in \mathsf{B}_{3m}\}_{j=1}^k$ such that
$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot f_{\mathsf{id}^*}(\tilde{\mathbf{z}}^j_{B^*}) \right) = \bar{\mathbf{A}} \cdot f_{\mathsf{id}}(\mathbf{z}_{B^*}) \equiv \sum_{\mathbf{u}_j \in B^*} \mathbf{u}_j \,.$$
The user further generates masking terms $\{\mathbf{r}^j_{B^*} \hookleftarrow \mathbb{Z}_q^{(2\ell+2)3m}\}_{j=1}^k$ such that the verifier can check the following expression used in the commitments once he transforms the policy into the DNF form.

$$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \left(\mathbf{r}^j_{B^*} + f_{\mathsf{id}^*}(\tilde{\mathbf{z}}^j_{B^*}) \right) \right) - \sum_{\mathbf{u}_j \in B^*} \mathbf{u}_j = \mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \mathbf{r}^j_{B^*} \right) .$$

- **Fake Credentials**: For the remaining disjunctive terms $C_i \neq C^*$ the user generates fake aggregated credentials, where $B_i \neq B^*$ defines the set of associated attributes. In particular, he samples uniform random vectors $\{\mathbf{d}^j_i \hookleftarrow \mathbb{Z}_q^{3m}\}_{j=1}^k$ such that

  $$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot f_{\mathsf{id}^*}(\mathbf{d}^j_i) \right) = \bar{\mathbf{A}} \cdot f_{\mathsf{id}}(\mathbf{d}_i) \equiv \sum_{\mathbf{u}_j \in B_i} \mathbf{u}_j .$$

  Analogously, the signer masks the fake credentials by uniform random vectors $\{\mathbf{r}^j_i \hookleftarrow \mathbb{Z}_q^{(2\ell+2)3m}\}_{j=1}^k$ for each disjunctive term. The verifier can check for the knowledge of the credentials

  $$\mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \left(\mathbf{r}^j_i + f_{\mathsf{id}^*}(\mathbf{d}^j_i) \right) \right) - \sum_{\mathbf{u}_j \in B_i} \mathbf{u}_j = \mathbf{A}_{\mathsf{ext}} \cdot \left(\sum_{j=1}^{k} \gamma_j \cdot \mathbf{r}^j_i \right) ,$$

  where $\sum_{j=1}^{k} \gamma_j \cdot \mathbf{d}^j_i = \mathbf{d}_i$ and $\sum_{j=1}^{k} \gamma_j \cdot \mathbf{r}^j_i = \mathbf{r}_i$.

The verifier validates the signature, i.e. the SPoK proof, by calling $\mathsf{Verify}_{\mathsf{E}}$ on the signature, policy and message. We note that the verifier first computes all aggregated attributes $\bar{\mathbf{u}}_i = \sum_{\mathbf{u}_j \in B_i} \mathbf{u}_j$ associated to C_i for $i \in [l]$ and then invokes the threshold verification sub-routine $\mathsf{Verify}_{\mathsf{T}}$. The security of the ABS scheme follows from the following two theorems.

Theorem 6 (Unforgeability). *If the NIZK system has special soundness and the attribute aggregation scheme is unforgeable, then our ABS scheme with expressive policies is unforgeable in the random oracle model.*

Theorem 7 (Anonymity). *If the NIZK system has statistical zero-knowledge, then our ABS scheme with expressive policies is anonymous in the random oracle model.*

The proofs of these statements exactly follow the same proof steps as for Theorem 2 and Theorem 3, since the scheme is an instance of the threshold ABS instantiated with the attribute aggregation scheme (which is basically the modified Boyen's signature scheme from Section 3.1 exploiting its aggregation property).

7.3 Efficiency and Comparison with the ABS Threshold Scheme.

The ABS scheme with expressive policies represents an instance of the $(1, B)$-threshold ABS scheme from Section 5 for $B = \{C_1, \ldots, C_l\}$, where l denotes the number of disjunctive terms in the policy $\psi = C_1 \vee \ldots \vee C_l$. In fact, the signature size is only dependent on the number l of disjunction terms, which is in most cases strictly smaller than the number n of related attributes. This reduces the communication cost, i.e. masking terms and permutations. Therefore, the ABS scheme with expressive policies is in general more efficient than its threshold counterpart in terms of signature size and hence also its performance. For instance, if there are two disjunctive terms only, where each is composed of $n/2$ different conjunctions, then the signature size depends only on 2 elements rather than the n attributes.

7.4 Traceability.

It is very straightforward to extend our constructions to allow for traceability by a given tracing authority. In fact, it is only required to encrypt the identity id^* by use of the public key of the tracing authority. This is accomplished, for instance, using the same tools as in [20,21,26] applied for group signatures. This is a standard technique to realize traceability. The statistical zero-knowledge argument of knowledge is hence extended by an additional term $\mathbf{c} = \mathbf{P}\mathbf{e} + (\mathbf{0}, \mathsf{id}^*)^\top$, i.e. the prover is required to provide a proof for the language

$$
\begin{aligned}
L = \{ \ & \mathsf{public} := \{\mathbf{A}, \{\mathbf{A}_i \in \mathbb{Z}_q^{n\times m}\}_{i=1}^{\ell}; \mathbf{u}_1, \ldots, \mathbf{u}_p, \mathbf{P}\}, \\
& \mathsf{witness} := \{\mathsf{id} \in \{0,1\}^{\ell}; \ \ \mathbf{z}_{j_1}, \ldots, \mathbf{z}_{j_t} \in \mathbb{Z}^{2m}; \mathbf{e} \in \mathbb{Z}^d\} : \\
& (\|\mathbf{z}_i\|_\infty \leq \gamma \ \wedge \ [\ \mathbf{A} \mid \mathbf{A}_0 + \sum_{j=1}^{\ell} \mathsf{id}_j \mathbf{A}_j \ ] \cdot \mathbf{z}_i = \mathbf{u}_i \bmod q \\
& \text{ for } i \in \{j_1, \ldots, j_t\}) \wedge (\mathbf{c} = \mathbf{P}\cdot\mathbf{e} + (\mathbf{0}, \mathsf{id})^\top \wedge \|\mathbf{e}\| \leq \beta) \ \}.
\end{aligned}
$$

8 Multi-Authority ABS Schemes

In many scenarios there exists not only one single attribute authority, but a number of different attribute authorities issuing credentials for various attributes. This also reflects real world scenarios, where a user is interacting within different domains such as universities or other institutions. However, some of the attribute authorities may be malicious or are even not aware of the other ones. In a multi-authority ABS scheme a signature trustee is setting up the various public parameters of the ABS scheme. This entity is not required to trust any of the attribute authorities. Our construction can naturally be extended to the multi-authority setting. We give a brief and informal overview of the key concepts.

- The user obtains key material $\mathsf{sk}_{\mathsf{id}}^{j}$ from attribute authority AA_j with public key $\mathsf{vk}_{\mathsf{AA}_j}$ and set of attributes $\mathsf{Attributes}_j$ for $j \in [k]$.
- For each threshold policy ψ_{j,t_j} related to authority j with threshold t_j the user can either separately generate an ABS signature for each policy or combine them within the SPoK, where the input to the commitment scheme is split into seperate parts for each authority.

  $$\mathbf{c}_i = (\text{ related to } \mathsf{AA}_1 \ || \ \dots \ || \text{ related to } \mathsf{AA}_k \ )$$

  If there is only one threshold policy ψ_t involving the attributes of various authorities, then the signer can proceed exactly as in Section 5 with the modification that the application of the public keys within the computation of $\mathbf{c}_1$ occurs in the same order as the occurrence in ψ_t so that the verifier is able to check $\mathbf{c}_1$.
- For expressive policies ψ one can produce independent ABS signatures on each policy or combine them into one SPoK, if the policies target each authority separately. However, if the expressive policy involves the attributes of different authorities, the user has to make sure that the DNF of ψ, i.e.

  $$\psi = C_1 \vee \ldots \vee C_l,$$

  contains only disjunctive terms C_i that target the attributes of only one authority, for instance C_1 is a conjunction of attributes from AA_3. In this case, we can directly apply the construction from Section 7, where the order of C_i and related public keys is reflected in the computation of

  $$\mathbf{c}_1 = (\ldots, \mathbf{A}_3 \cdot \mathbf{r}_3, \ldots, \mathbf{A}_k \cdot \mathbf{r}_k, \ldots)$$

  For policies, where any of the $\bigwedge$-policies ψ_{C_i} involve more than one party, it is not immediately possible to apply one of the proposed constructions, since they require to aggregate all the related credentials of a policy ψ_{C_i}.

Acknowledgements

We would like to thank Essam Ghadafi (UCL) for useful discussions.

References

1. Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). pages 99–108, 1996.
2. W. Banaszczyk. Inequalities for convex bodies and polar reciprocal lattices in r^n. *Discrete and Computational Geometry*, 13(1):217–231, 1995.
3. Rakesh Bobba, Omid Fatemieh, Fariba Khan, Carl A. Gunter, and Himanshu Khurana. Using attribute-based access control to enable attribute-based messaging. In *ACSAC. IEEE CS*, pages 403–413, 2006.
4. Xavier Boyen. Lattice mixing and vanishing trapdoors: A framework for fully secure short signatures and more. pages 499–517, 2010.
5. Xavier Boyen. Attribute-based functional encryption on lattices. *Theory of Cryptography: 10th Theory of Cryptography Conference, TCC 2013*, pages 122–142, 2013.
6. Jan Camenisch, Gregory Neven, and Markus Rückert. Fully anonymous attribute tokens from lattices. *Security and Cryptography for Networks: 8th International Conference, SCN 2012*, pages 57–75, 2012.
7. David Chaum and Eugène Van Heyst. Group signatures. EUROCRYPT, pages 257–265. Springer-Verlag, 1991.
8. Shantian Cheng, Khoa Nguyen, and Huaxiong Wang. Policy-based signature scheme from lattices. *Designs, Codes and Cryptography*, pages 1–32, 2015.
9. Özgür Dagdelen, Marc Fischlin, and Tommaso Gagliardoni. *The Fiat–Shamir Transformation in a Quantum World*, pages 62–81. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
10. Ali El Kaafarani, Liqun Chen, Essam Ghadafi, and James Davenport. Attribute-based signatures with user-controlled linkability. In *CANS*, pages 256–269. Springer, 2014.
11. Ali El Kaafarani, Essam Ghadafi, and Dalia Khader. Decentralized traceable attribute-based signatures. In *CT-RSA*, pages 327–348. Springer, 2014.
12. Alex Escala, Javier Herranz, and Paz Morillo. Revocable attribute-based signatures with adaptive security in the standard model. In *AFRICACRYPT*, pages 224–241. Springer, 2011.
13. Keith B. Frikken, Jiangtao Li, and Mikhail J. Atallah. Trust negotiation with hidden credentials, hidden policies, and policy cycles. In *NDSS*, pages 157–172, 2006.
14. Martin Gagné, Shivaramakrishnan Narayan, and Reihaneh Safavi-Naini. Short pairing-efficient threshold-attribute-based signature. In *Pairing*, volume 7708, pages 295–313. Springer, 2013.
15. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. pages 197–206, 2008.
16. Essam Ghadafi. Stronger security notions for decentralized traceable attribute-based signatures and more efficient constructions. In Kaisa Nyberg, editor, *CT-RSA*, pages 391–409. Springer, 2015.
17. S. Dov Gordon, Jonathan Katz, and Vinod Vaikuntanathan. A group signature scheme from lattice assumptions. *Advances in Cryptology - ASIACRYPT 2010*, pages 395–412, 2010.
18. Javier Herranz, Fabien Laguillaumie, Benoît Libert, and Carla Ràfols. Short attribute-based signatures for threshold predicates. In *CT-RSA*, volume 7178, pages 51–67. Springer, 2012.

19. Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. pages 372–389, 2008.
20. Fabien Laguillaumie, Adeline Langlois, Benoît Libert, and Damien Stehlé. Lattice-based group signatures with logarithmic signature size. *Advances in Cryptology - ASIACRYPT 2013: 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part II*, pages 41–61, 2013.
21. Adeline Langlois, San Ling, Khoa Nguyen, and Huaxiong Wang. Lattice-based group signature scheme with verifier-local revocation. *Public-Key Cryptography – PKC 2014: 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings*, pages 345–361, 2014.
22. Jin Li, Man Ho Au, Willy Susilo, Dongqing Xie, and Kui Ren. Attribute-based signature and its applications. In *ASIACCS*, pages 60–69. ACM, 2010.
23. Jin Li and Kwangjo Kim. Attribute-based ring signatures. Cryptology ePrint Archive, Report 2008/394, 2008. `http://eprint.iacr.org/2008/394`.
24. Benoît Libert, San Ling, Khoa Nguyen, and Huaxiong Wang. Zero-knowledge arguments for lattice-based accumulators: Logarithmic-size ring signatures and group signatures without trapdoors. *Advances in Cryptology – EUROCRYPT 2016*, pages 1–31, 2016.
25. San Ling, Khoa Nguyen, Damien Stehlé, and Huaxiong Wang. Improved zero-knowledge proofs of knowledge for the ISIS problem, and applications. pages 107–124, 2013.
26. San Ling, Khoa Nguyen, and Huaxiong Wang. Group signatures from lattices: Simpler, tighter, shorter, ring-based. pages 427–449, 2015.
27. Hemanta K Maji, Manoj Prabhakaran, and Mike Rosulek. Attribute-based signatures. In *CT-RSA*, pages 376–392. Springer, 2011.
28. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. pages 700–718, 2012.
29. Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. pages 372–381, 2004.
30. Tatsuaki Okamoto and Katsuyuki Takashima. Efficient attribute-based signatures for non-monotone predicates in the standard model. In *PKC*, pages 35–52. Springer, 2011.
31. Tatsuaki Okamoto and Katsuyuki Takashima. Decentralized attribute-based signatures. In *PKC*, pages 125–142. Springer, 2013.
32. Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. In *ASIACRYPT*, pages 552–565. Springer, 2001.
33. Siamak F. Shahandashti and Reihaneh Safavi-Naini. Threshold attribute-based signatures and their application to anonymous credential systems. In *AFRICACRYPT*, pages 198–216. Springer, 2009.
34. Dominique Unruh. Non-interactive zero-knowledge proofs in the quantum random oracle model. pages 755–784, 2015.
35. Miguel Urquidi, Dalia Khader, Jean Lancrenon, and Liqun Chen. Attribute-based signatures with controllable linkability. In *INTRUST*, volume 9565, pages 114–129. Springer, 2015.
36. Brent R. Waters. Efficient identity-based encryption without random oracles. pages 114–127, 2005.