The preliminary version of this paper appeared in the proceedings of *"The 3rd IEEE International Workshop on Big Data and IoT Security in Smart Computing - IEEE BITS 2019 -"* (co-held with *"IEEE SMARTCOMP 2019"*) under the same title. This is the full version.

Decentralized Multi-authority Anonymous Authentication for Global Identities with Non-interactive Proofs

Hiroaki Anada[1]

Department of Information Security, University of Nagasaki
W408, 1-1-1, Manabino, Nagayo-cho, Nishisonogi-gun, Nagasaki, 851-2195 Japan
anada@sun.ac.jp

August 23, 2019

Abstract. We propose a decentralized multi-authority anonymous authentication scheme in which a prover and a verifier are non-interactive. We give two security definitions; resistance against collusion attacks that cause misauthentication, and anonymity for privacy protection. Then we give a construction of our scheme under a principle of "commit-to-ID". We employ two building blocks; the structure-preserving signature scheme and the Groth-Sahai non-interactive proof system, the both of which are based on bilinear groups. We give security proofs in the standard model, which reduce the security of our scheme to the security of the building blocks.

Keywords: anonymous authentication, attribute, collusion attack, identity, non-interactive

1 Introduction

Privacy protection in cyber-physical space is a function that should be pursued in authentication. The growth of companies in the areas of the IT infrastructures made protecting privacy of vital importance for involved users in recent years because we use search engines, digital devices, social networking services and e-shopping services everyday. Considering the change of our life and business going the cyber-physical space, one of the critical aspects that our future authentication framework should attain is anonymous authentication via *attributes*. For example, connected-to-the-internet vehicles, bicycles and even human beings with embedded devices will use plural services like GPS, availability of nearby places, disclosed data of other movable entities, and suitable options of business strategy. There is no need of identity information of the user, but instead, the user should be authenticated in anonymous way using her attribute credentials issued beforehand by independent administration authorities related to the service providers. Another aspect is optimization of a service based on plural other services. That is, there will be a compound service model which involves the independent administration authorities *at a time*. For example, location data from GPS, availability information of nearby places and disclosed data of other entities can be thrown into the input of smart computation of optimizing business strategies.

However, there is a threat on such a framework of anonymous authentication using plural independent attribute credentials; *collusion attack*. That is, malicious users of different identities bring together their attribute credentials. They try to make a verifier accept anonymously using the merged attribute credentials. Here the very anonymity is a critical potential drawback from the view point of the collusion attack that causes misauthentication.

1.1 Related Work and Our Contribution

In this paper, we propose a decentralized multi-authority anonymous authentication scheme to resolve the above problem. Our scheme is a special case of a decentralized multi-authority attribute-based signature scheme (DMA-ABS). Okamoto-Takashima [OT13] proposed a state-of-the-art DMA-ABS, which can treat general non-monotone access structures, while our scheme treats only the all-AND structure. Nonetheless, a feature of our scheme is that, when a prover wants the authorities to issue private secret keys as attribute credentials, the authorities simply *generate digital signatures* on her global identity group element. This feature is useful when her global identity group element is easy to be validated in registration phase by the authorities. The second feature of our scheme is that the authorities are independent each other, while the computational amount and the proof length of a prover grows linearly to the number of authorities involved.

Anonymous credential systems (ACS) [CL01] and accumulators [CL02] are also comparable with our scheme. Sadiah et al. [SNBF17] proposed a state-of-the-art ACS with accumulators, which can treat general monotone access structures, while our scheme treats only the all-AND formula. Nonetheless, our scheme has the two features to which the ACS does not attain. Camenisch et al. [CDHK15,CDT19] proposed a similar scheme which has the two features, but their scheme does not treat collusion resistance that is our first aim.

There is also previous work by Anada-Arita [AA18a,AA18b] which shares the two features. However, their authentication scheme is interactive between a prover and a verifier in accordance with the Σ-protocol [Dam10], whereas our scheme is non-interactive based on the Groth-Sahai proof system [GS08] yielding better availability for applications which need quick authentication.

1.2 Overview of Our Construction and Security Proofs

In Section 6, we will construct a decentralized multi-authority anonymous authentication scheme with non-interactive proofs (NI-DMA-A-AUTH). There we employ two building blocks. One is the structure-preserving signature scheme [AFG$^+$10,AFG$^+$16] (see Section 2.2). Each decentralized authority indexed by 'a' issues a private secret key $\mathrm{sk}^a_{\mathtt{gid}}$ by signing a global identity group element $\mathtt{gid}$ of a prover. Here $\mathtt{gid}$ is a group element of one of the source groups of bilinear groups. As a remark, a realistic constraint is that the authorities have to refer to a common set of public parameters pp, as is usual in the case of NIST Standard (like NIST.FIPS.186-4 [NIS13]). The other building block is the non-interactive commit-and-prove scheme of the fine-tuned Groth-Sahai proof system [GS08,EG14]. We give a description of a version adopted to the case of proving knowledge of the structure-preserving signatures (see Section 3). Our construction is under a principle of "commit-to-ID". That is, in the commit-phase the prover commits to an identity group element $\mathtt{gid}$ ("global identity"). She also commits to the components of the structure-preserving signatures, which are also group elements. In the prove-phase the prover generates a proof π of knowing a solution of the verification equation system of the structure-preserving signatures. Here the common single commitment c_0 to $\mathtt{gid}$ works for proving knowledge of *bundled witnesses* (see "*bundled* languages" in Section 4). Thus, the collusion attack becomes impossible due to the binding property of the commitment c_0 because c_0 is common in the verification equation system. In other words, our construction builds on the "plug-in" mechanism of commitments in generating a proof π in the Groth-Sahai proof system (see [GS08]).

In Section 5, after giving the syntax of our NI-DMA-A-AUTH, we give two security definitions. One is resistance against collusion attacks that cause misauthentication, and the other is anonymity for privacy protection. There we capture the requirements described above. In Section 6, security proofs for the above construction are given in the standard model. The resistance against collusion attacks is due to knowledge extraction property of the Groth-Sahai proof system and existential unforgeability of the structure-preserving signature scheme. The anonymity is due to perfectly hiding property of commitments and perfectly witness indistinguishable property of proofs of the Groth-Sahai proof system, where the both properties hold in the simulation mode of the dual mode commitment.

2 Preliminaries

The set of natural numbers is denoted by $\mathbb{N}$. We put $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$. The residue class ring of integers modulo a prime number p is denoted by $\mathbb{Z}_p$. The security parameter is denoted by λ, where $\lambda \in \mathbb{N}$. A probability P is said to be negligible in λ if for any given positive polynomial $\mathrm{poly}(\lambda)$ $P < 1/\mathrm{poly}(\lambda)$ for sufficiently large

$\lambda \in \mathbb{N}$. Two probabilities P and Q are said to be computationally indistinguishable if $|P - Q|$ is negligible in λ, which is denoted as $P \approx_c Q$. A uniform random sampling of an element a from a set S is denoted as $a \in_U S$. When a probabilistic algorithm A with an input a and a randomness r on a random tape returns z, we denote it as $z \leftarrow A(a; r)$ St is the inner state of the concerned algorithm.

2.1 Bilinear Groups [GPS08,EG14]

Let $\mathcal{G}$ be a generation algorithm of bilinear groups [GPS08]: $\mathcal{G}(1^\lambda) \to (p, \hat{\mathbb{G}}, \check{\mathbb{H}}, \mathbb{T}, e, \hat{G}, \check{H})$. Here p is a prime number of bit-length λ, $\hat{\mathbb{G}}, \check{\mathbb{H}}$ and $\mathbb{T}$ are cyclic groups of order p, and $\hat{G}$ and $\check{H}$ are generators of $\hat{\mathbb{G}}$ and $\check{\mathbb{H}}$, respectively. We denote operations in $\hat{\mathbb{G}}$, $\check{\mathbb{H}}$ and $\mathbb{T}$ multiplicatively. e is a map $e : \hat{\mathbb{G}} \times \check{\mathbb{H}} \to \mathbb{T}$ with the following two properties:

- Non-degeneracy : $e(\hat{G}, \check{H}) \neq 1_{\mathbb{T}}$
- Bilinearity : $\forall a \in \mathbb{Z}_p, \forall b \in \mathbb{Z}_p, \forall \hat{X} \in \hat{\mathbb{G}}, \forall \check{Y} \in \check{\mathbb{H}}, e(\hat{X}^a, \check{Y}^a) = e(\hat{X}, \check{Y})^{ab}$.

Hereafter we denote an element in $\hat{\mathbb{G}}$ and $\check{\mathbb{H}}$ with hat ' $\hat{}$ ' and check ' $\check{}$ ' , respectively.

2.2 Structure-Preserving Signature Scheme [AFG$^+$10,AFG$^+$16]

The structure-preserving signature scheme Sig is a digital signature scheme based on bilinear groups, in which a message is a vector whose entries belong to one of the two source groups $\hat{\mathbb{G}}$ and $\check{\mathbb{H}}$, and a signature is a vector whose entries belong to $\hat{\mathbb{G}}$ and $\check{\mathbb{H}}$. Based on Abe et al. [AFG$^+$10,AFG$^+$16], we survey the four PPT algorithms of the structure-preserving signature scheme $\mathsf{Sig} = (\mathsf{Sig.Setup}, \mathsf{Sig.KG}_{pp}, \mathsf{Sig.Sign}_{pp}, \mathsf{Sig.Vrf}_{pp})$.
$\mathsf{Sig.Setup}(1^\lambda) \to pp$. On input the security parameter 1^λ, this PPT algorithm executes the generation algorithm of bilinear groups, and it puts the output as a set of public parameters: $\mathcal{G}(1^\lambda) \to (p, \hat{\mathbb{G}}, \check{\mathbb{H}}, \mathbb{T}, e, \hat{G}, \check{H}) =: pp$. It returns pp.
$\mathsf{Sig.KG}_{pp}() \to (\mathrm{PK}, \mathrm{SK})$. Based on the set of public parameters pp, this PPT algorithm generates a signing key SK and the corresponding public key PK as follows: $\hat{G}_u \in_U \hat{\mathbb{G}}$, $\gamma_1, \delta_1 \in_U \mathbb{Z}_p^*$, $\hat{G}_1 := \hat{G}^{\gamma_1}, \hat{G}_{u,1} := \hat{G}_u^{\delta_1}$. $\gamma_z, \delta_z \in_U \mathbb{Z}_p^*$, $\hat{G}_z := \hat{G}^{\gamma_z}, \hat{G}_{u,z} := \hat{G}_u^{\delta_z}$. $\alpha, \beta \in_U \mathbb{Z}_p^*$, $(\hat{A}_i, \check{A}_i)_{i=0}^1 \leftarrow \mathsf{Extend}(\hat{G}, \check{H}^\alpha)$, $(\hat{B}_i, \check{B}_i)_{i=0}^1 \leftarrow \mathsf{Extend}(\hat{G}_u, \check{H}^\beta)$ (for Extend, see [AFG$^+$16]). It puts $\mathrm{PK} := (\hat{G}_z, \hat{G}_{u,z}, \hat{G}_u, \hat{G}_1, \hat{G}_{u,1}, (\hat{A}_i, \check{A}_i, \hat{B}_i, \check{B}_i)_{i=0}^1)$ and $\mathrm{SK} := (\alpha, \beta, \gamma_z, \delta_z, \gamma_1, \delta_1)$. It returns $(\mathrm{PK}, \mathrm{SK})$.
$\mathsf{Sig.Sign}_{pp}(\mathrm{PK}, \mathrm{SK}, m) \to \sigma$. On input the public key PK, the secret key SK and a message $m = \check{M} \in \check{\mathbb{H}}$, this PPT algorithm generates a signature σ as follows.

$$\zeta, \rho, \tau, \phi, \omega \in_U \mathbb{Z}_p, \quad \check{Z} := \check{H}^\zeta, \check{R} := \check{H}^{\alpha - \rho\tau - \gamma_z\zeta} \check{M}^{-\gamma_1}, \hat{S} := \hat{G}^\rho, \check{T} := \check{H}^\tau,$$
$$\check{U} := \check{H}^{\beta - \phi\omega - \delta_z\zeta} \check{M}^{-\delta_1}, \hat{V} := \hat{G}_u^\phi, \check{W} := \check{H}^\omega.$$

It returns $\sigma := (\check{Z}, \check{R}, \hat{S}, \check{T}, \check{U}, \hat{V}, \check{W})$.
$\mathsf{Sig.Vrf}_{pp}(\mathrm{PK}, m, \sigma) \to d$. On input the public key PK, a message $m = \check{M} \in \check{\mathbb{H}}$ and a signature $\sigma = (\check{Z}, \check{R}, \hat{S}, \check{T}, \check{U}, \hat{V}, \check{W})$, this deterministic algorithm checks whether the following verification equation system holds or not.

$$e(\hat{G}_z, \check{Z}) e(\hat{G}, \check{R}) e(\hat{S}, \check{T}) e(\hat{G}_1, \check{M}) e(\hat{A}_0, \check{A}_0)^{-1} e(\hat{A}_1, \check{A}_1)^{-1} = 1_{\mathbb{T}}, \text{ and} \tag{1}$$
$$e(\hat{G}_{u,z}, \check{Z}) e(\hat{G}_u, \check{U}) e(\hat{V}, \check{W}) e(\hat{G}_{u,1}, \check{M}) e(\hat{B}_0, \check{B}_0)^{-1} e(\hat{B}_1, \check{B}_1)^{-1} = 1_{\mathbb{T}}. \tag{2}$$

It returns a boolean decision d.

The correctness should hold for the scheme Sig: For any security parameter 1^λ, any set of public parameters $pp \leftarrow \mathsf{Sig.Setup}(1^\lambda)$ and any message $m = \check{M} \in \check{\mathbb{H}}$, $\Pr[d = 1 \mid (\mathrm{PK}, \mathrm{SK}) \leftarrow \mathsf{Sig.KG}_{pp}(), \sigma \leftarrow \mathsf{Sig.Sign}_{pp}(\mathrm{PK}, \mathrm{SK}, m), d \leftarrow \mathsf{Sig.Vrf}_{pp}(\mathrm{PK}, m, \sigma)] = 1$.

Adaptive chosen-message attack of an existential forgery on the scheme Sig by a forger algorithm $\mathbf{F}$ is defined by the following algorithm of experiment.

$$
\begin{aligned}
&\mathsf{Exp}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(1^\lambda):\\
&\quad pp \leftarrow \mathsf{Sig.Setup}(1^\lambda), (\mathrm{PK},\mathrm{SK}) \leftarrow \mathsf{Sig.KG}_{pp}()\\
&\quad (m^*,\sigma^*) \leftarrow \mathbf{F}^{\mathbf{SignO}_{pp}(\mathrm{PK},\mathrm{SK},\cdot)}(pp,\mathrm{PK})\\
&\quad \text{If } m^* \notin \{m_j\}_{1\le j\le q_\mathrm{s}} \text{ and } \mathsf{Sig.Vrf}_{pp}(\mathrm{PK},m^*,\sigma^*)=1,\\
&\qquad \text{then Return } \textsc{Win} \text{ else Return } \textsc{Lose}
\end{aligned}
$$

In the experiment, $\mathbf{F}$ issues a signing query to its signing oracle $\mathbf{SignO}_{pp}(\mathrm{PK},\mathrm{SK},\cdot)$ by sending a message m_j at most q_s times ($1 \le j \le q_\mathrm{s}$). As a reply, $\mathbf{F}$ receives a valid signature σ_j on m_j. After receiving replies, $\mathbf{F}$ returns a pair of a message and a signature (m^*,σ^*). A restriction is imposed on the algorithm $\mathbf{F}$: The set of queried messages $\{m_j\}_{1\le j\le q_\mathrm{s}}$ should not contain the message m^*. The advantage of $\mathbf{F}$ over Sig is defined as $\mathbf{Adv}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(\lambda) := \Pr[\mathsf{Exp}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(1^\lambda) \text{ returns } \textsc{Win}]$. The scheme Sig is said to be *existentially unforgeable against adaptive chosen-message attacks (EUF-CMA)* if for any PPT algorithm $\mathbf{F}$ and any q_s bounded by a polynomial in λ, the advantage $\mathbf{Adv}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(\lambda)$ is negligible in λ. The structure-preserving signature scheme [AFG+10,AFG+16] is known to be EUF-CMA under the q-SFP assumption.

3 Non-interactive Commit-and-Prove Scheme for Structure-Preserving Signatures

In this section, we give a description of the non-interactive commit-and-prove scheme of the fine-tuned Groth-Sahai proof system [EG14] adapted to the case of our specific group-dependent language; that is, the language of the verification equation system of the structure-preserving signature.

A commit-and-prove scheme CmtPrv consists of six PPT algorithms: $\mathsf{CmtPrv} = (\mathsf{CmtPrv.Setup}, \mathsf{Cmt}_{pp} = (\mathsf{Cmt.KG}_{pp}, \mathsf{Cmt.Com}_{pp}, \mathsf{Cmt.Vrf}_{pp}), \mathit{\Pi}_{pp} = (\mathsf{P}_{pp}, \mathsf{V}_{pp}))$.

3.1 Commitment Part

The commitment part $(\mathsf{CmtPrv.Setup}, \mathsf{Cmt}_{pp})$ is described as follows.

• $\mathsf{CmtPrv.Setup}(1^\lambda) \to pp$. On input the security parameter 1^λ, this PPT algorithm executes a generation algorithm of bilinear groups, and it puts the output as a set of public parameters: $\mathcal{G}(1^\lambda) \to (p, \hat{\mathbb{G}}, \check{\mathbb{H}}, \mathbb{T}, e, \hat{G}, \check{H}) =: pp$. It returns pp.

• $\mathsf{Cmt.KG}_{pp}(\mathtt{mode}) \to key$. On input a string $\mathtt{mode}$, this PPT algorithm generates a key. If $\mathtt{mode} = \mathtt{nor}$, then $key = ck$ which is a commitment key. If $\mathtt{mode} = \mathtt{ext}$, then $key = (ck, xk)$ which is a pair of ck and an extraction key xk. If $\mathtt{mode} = \mathtt{sim}$, then $key = (ck, tk)$ which is a pair of ck and a trapdoor key tk. It returns key.

We put $pp := (pp, ck)$ because the commitment key ck is treated as a public parameter.

• $\mathsf{Cmt.Com}_{pp}(w; r) \to (c, r)$. On input a message w which may be a vector, this PPT algorithm generates a commitment c with a randomness r. It returns (c, r). If w is a vector $w = (w_0, \ldots, w_{n-1})$ (for some $n \in \mathbb{N}$ bounded by a polynomial in λ), then c and r are also vectors of the same number of components: $c = (c_0, \ldots, c_{n-1})$ and $(r_0, \ldots, r_{n-1})$, respectively. Note also that computation is executed in *componentwise way*; c_i is generated from w_i and r_i, $i = 0, \ldots, n-1$.

• $\mathsf{Cmt.Vrf}_{pp}(c, w, r) \to d$. On input a commitment c, a message w and a verification key r, this deterministic algorithm generates a boolean decision d. It returns d.

The commitment part $(\mathsf{CmtPrv.Setup}, \mathsf{Cmt}_{pp})$ of the Groth-Sahai proof system has the following four properties.

Definition 1 (Correctness [GS08]) *A commitment scheme Cmt_{pp} is said to be correct if it satisfies the following condition: For any security parameter 1^λ, any set of public parameters $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$, any commitment key $ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{mode})$ where $\mathtt{mode} = \mathtt{nor}$ or $\mathtt{ext}$ or $\mathtt{sim}$, and any message w,*

$$\Pr[d = 1 \mid (c, r) \leftarrow \mathsf{Cmt.Com}_{pp}(w), d \leftarrow \mathsf{Cmt.Vrf}_{pp}(c, w, r)] = 1.$$

Definition 2 (Dual Mode [GS08]) *A commitment scheme* Cmt_{pp} *is said to be dual mode if it satisfies the following condition: For any security parameter* 1^λ*, any set of public parameters* $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$ *and any* PPT *algorithm* $\mathbf{A}$*,*

$$\begin{aligned} &\Pr[\mathbf{A}(pp, ck) = 1 \mid ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})] \\ =\ &\Pr[\mathbf{A}(pp, ck) = 1 \mid (ck, xk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{ext})], \qquad (3) \\ &\Pr[\mathbf{A}(pp, ck) = 1 \mid ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})] \\ \approx_c\ &\Pr[\mathbf{A}(pp, ck) = 1 \mid (ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim})]. \qquad (4) \end{aligned}$$

The computational indistinguishability (4) is equivalent to the following: For any security parameter 1^λ, for any set of public parameters $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$ and any PPT algorithm $\mathbf{A}$, the advantage $\mathbf{Adv}^{\text{ind-dual}}_{\mathsf{Cmt}_{pp},\mathbf{A}}(\lambda)$ of $\mathbf{A}$ over Cmt_{pp} defined by the difference below is negligible in λ:

$$\begin{aligned} \mathbf{Adv}^{\text{ind-dual}}_{\mathsf{Cmt}_{pp},\mathbf{A}}(\lambda) \stackrel{\text{def}}{=} |\ &\Pr[\mathbf{A}(pp, ck) = 1 \mid ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})] \\ &- \Pr[\mathbf{A}(pp, ck) = 1 \mid (ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim})]|. \qquad (5) \end{aligned}$$

The indistinguishability holds, for example, for an instance of the Groth-Sahai proof system under the SXDH assumption [GS08,EG14].

Definition 3 (Perfect Binding [GS08]) *A commitment scheme* Cmt_{pp} *is said to be perfectly binding if it satisfies the following condition for some unbounded algorithm* $\mathsf{Cmt.Open}_{pp}$*: For any security parameter* 1^λ*, any set of public parameters* $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$*, any commitment key* $ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})$ *and any message* w*,*

$$\Pr[w = w' \mid (c, r) \leftarrow \mathsf{Cmt.Com}_{pp}(w; r), w' \leftarrow \mathsf{Cmt.Open}_{pp}(c)] = 1.$$

Definition 4 (Perfect Hiding [GS08]) *A commitment scheme* Cmt_{pp} *is said to be perfectly hiding if it satisfies the following condition: For any security parameter* 1^λ*, any set of public parameters* $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$*, any commitment key* ck *s.t.* $(ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim})$ *and any* PPT *algorithm* $\mathbf{A}$*,*

$$\begin{aligned} &\Pr[\mathbf{A}(St, c) = 1 \mid (w, w', St) \leftarrow \mathbf{A}(pp, ck, tk), (c, r) \leftarrow \mathsf{Cmt.Com}_{pp}(w)] \\ =\ &\Pr[\mathbf{A}(St, c') = 1 \mid (w, w', St) \leftarrow \mathbf{A}(pp, ck, tk), (c', r') \leftarrow \mathsf{Cmt.Com}_{pp}(w')]. \qquad (6) \end{aligned}$$

3.2 Proof Part

The proof-part $(\mathsf{CmtPrv.Setup}, \mathit{\Pi}_{pp})$ is described as follows. Let $\mathcal{CK}_{pp}$ denote the set of commitment keys, $\mathcal{X}_{pp}$ denote the set of coefficients of the verification equation system (1) and (2), and $\mathcal{W}_{pp}$ denote the set of the pairs of messages and signatures for some $x \in \mathcal{X}_{pp}$:

$$\begin{aligned} \mathcal{CK}_{pp} &= \{ck \mid ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{mode}) \text{ for } \mathtt{mode} = \mathtt{nor} \text{ or } \mathtt{ext} \text{ or } \mathtt{sim}\}, \\ \mathcal{X}_{pp} &= \{x \mid (\mathrm{PK}, \mathrm{SK}) \leftarrow \mathsf{Sig.KG}_{pp}(), x = \mathrm{PK}\}, \\ \mathcal{W}_{pp} &= \{w \mid w = (w_0, w_1, \ldots, w_7) \in \check{\mathbb{H}}^3 \times \hat{\mathbb{G}} \times \check{\mathbb{H}}^2 \times \hat{\mathbb{G}} \times \check{\mathbb{H}} \\ &\qquad \text{s.t. (1) and (2) hold for } \exists x \in \mathcal{X}, \\ &\qquad w_0 = m = \check{M},\ (w_1, \ldots, w_7) = \sigma = (\check{Z}, \check{R}, \hat{S}, \check{T}, \check{U}, \hat{V}, \check{W})\}. \end{aligned}$$

Then we define the following ternary relation R_{pp}.

$$\begin{aligned} R_{pp} \stackrel{\text{def}}{=} \{(ck, x, w) \in \mathcal{CK}_{pp} \times \mathcal{X}_{pp} \times \mathcal{W}_{pp} \mid\ &w \text{ can be committed by } \mathsf{Cmt.Com}_{pp} \text{ under } ck \\ &\text{and (1) and (2) hold for } (x, w)\}. \end{aligned}$$

A group-dependent language $L_{pp,ck}$ parametrized by $ck \in \mathcal{CK}$ is defined as follows.

$$L_{pp,ck} \stackrel{\text{def}}{=} \{x \in \mathcal{X}_{pp} \mid \exists w \in \mathcal{W}_{pp} \text{ s.t. } (ck, x, w) \in R_{pp}\}.$$

We put $pp := (pp, ck)$ because the commitment key ck is treated as a public parameter.
• $\mathsf{P}_{pp}(x, c, w, r) \to \pi$. On input a statement x, a commitment c, a witness w and a randomness r which was used to generate a commitment c, this PPT algorithm executes the proof-generation algorithm of the Groth-Sahai proof system to obtain a proof π (see [EG14] for the details). It returns π.
• $\mathsf{V}_{pp}(x, c, \pi) \to d$. On input a statement x, a commitment c and a proof π, this deterministic algorithm executes the verification algorithm of the Groth-Sahai proof system to obtain a boolean decision d (see [EG14] for the details). It returns d.

The proof-part $(\mathsf{CmtPrv.Setup}, \Pi_{pp})$ of the Groth-Sahai proof system have the following four properties.

Definition 5 (Perfect Correctness [GS08]) *A commit-and-prove scheme CmtPrv is said to be perfectly correct if it satisfies the following condition: For any security parameter 1^λ, any set of public parameters $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$, any commitment key $ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{mode})$ where $\mathtt{mode} = \mathtt{nor}$ or $\mathtt{ext}$ or $\mathtt{sim}$, and any* PPT *algorithm $\mathbf{A}$,*

$$\begin{aligned}\Pr[\mathsf{V}_{pp}(x, c, \pi) = 1 \text{ if } (ck, x, w) \in R_{pp} \mid \\ (x, w) \leftarrow \mathbf{A}(pp), (c, r) \leftarrow \mathsf{Cmt.Com}_{pp}(w), \\ \pi \leftarrow \mathsf{P}_{pp}(x, c, w, r)] = 1.\end{aligned}$$

Definition 6 (Perfect Soundness [GS08]) *A commit-and-prove scheme CmtPrv is said to be perfectly sound if it satisfies the following condition for some unbounded algorithm $\mathsf{Cmt.Open}_{pp}$: For any security parameter 1^λ, any set of public parameters $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$, any commitment key $ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})$ and any* PPT *algorithm $\mathbf{A}$,*

$$\begin{aligned}\Pr[\mathsf{V}_{pp}(x, c, \pi) = 0 \text{ or } (ck, x, w) \in R_{pp} \mid \\ (x, c, \pi) \leftarrow \mathbf{A}(pp), w \leftarrow \mathsf{Cmt.Open}_{pp}(c)] = 1.\end{aligned}$$

Let $\mathcal{C}_{ck}$ be the set of commitments under ck to some message w.

Definition 7 (Perfect Knowledge Extraction [GS08]) *A commit-and-prove scheme CmtPrv is said to be perfectly knowledge extractable if it satisfies the following condition for some* PPT *algorithm $\mathsf{Cmt.Ext}_{pp}$: For any security parameter 1^λ, any set of public parameters $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$, any commitment key $(ck, xk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{ext})$ and any* PPT *algorithm $\mathbf{A}$,*

$$\Pr[c \notin \mathcal{C}_{ck} \text{ or } \mathsf{Cmt.Ext}_{pp}(xk, c) = \mathsf{Cmt.Open}_{pp}(c) \mid c \leftarrow \mathbf{A}(pp, ck, xk)] = 1.$$

Definition 8 (Composable Witness-Indistinguishability [GS08]) *A commit-and-prove scheme CmtPrv is said to be composably witness-indistinguishable if it satisfies the following condition: For any security parameter 1^λ, any set of public parameters $pp \leftarrow \mathsf{CmtPrv.Setup}(1^\lambda)$ and any* PPT *algorithm $\mathbf{A}$,*

$$\begin{aligned}&\Pr[\mathbf{A}(pp, ck) = 1 \mid ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})] \\ \approx_c\ &\Pr[\mathbf{A}(pp, ck) = 1 \mid (ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim})], \text{ and} \\ &\Pr[(ck, x, w), (ck, x, w') \in R_{pp} \text{ and } \mathbf{A}(St, \pi) = 1 \mid (ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim}), pp := (pp, ck), \\ &\quad (x, w, w', St) \leftarrow \mathbf{A}^{\mathsf{Cmt.Com}_{pp}(\cdot)}(pp, ck, tk), (c, r) \leftarrow \mathsf{Cmt.Com}_{pp}(w), \pi \leftarrow \mathsf{P}_{pp}(x, c, w, r)] \\ =\ &\Pr[(ck, x, w), (ck, x, w') \in R_{pp} \text{ and } \mathbf{A}(St, \pi') = 1 \mid (ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim}), pp := (pp, ck), \\ &\quad (x, w, w', St) \leftarrow \mathbf{A}^{\mathsf{Cmt.Com}_{pp}(\cdot)}(pp, ck, tk), (c', r') \leftarrow \mathsf{Cmt.Com}_{pp}(w'), \pi' \leftarrow \mathsf{P}_{pp}(x, c', w', r')]. \qquad (7)\end{aligned}$$

4 Bundled Languages

In this section, we define a general notion of bundled languages for the case of the group-dependent languages [GS08]. Intuitively, the notion determines a subset of the Cartesian product of a language by the condition that the corresponding witnesses have a fixed number of common components in the former part.

Definition 9 (k-bundled languages) *Let pp be a given set of public parameters including a commitment key ck. Let L_{pp} be a group-dependent language. For a polynomially bounded integer q, let A be the set of indices: $A := \{1, \ldots, q\}$. For $a \in A$, put $L^a_{pp} := L_{pp}$. Let $k \in \mathbb{N}_0$. Suppose that, for pp, a witness w^a consists of n components: $w^a = (w^a_0, \ldots, w^a_{n-1})$. The k-bundled product $\prod^{k\text{-}bnd}_{a\in A} L^a_{pp}$ of languages $L^a_{pp}, a \in A$, is defined as follows.*

$$\prod_{a\in A}^{k\text{-}bnd} L^a_{pp} \stackrel{def}{=} \{(x^a)^{a\in A} \in \prod_{a\in A} L^a_{pp} \mid \exists w_0, \ldots, w_{k-1}, \forall a \in A, \exists w^a_k, \ldots, w^a_{n-1}, \\ w^a = (w_0, \ldots, w_{k-1}, w^a_k, \ldots, w^a_{n-1}), (ck, x^a, w^a) \in R\}. \quad (8)$$

We call $\prod^{k\text{-}bnd}_{a\in A} L^a_{pp}$ k-bundled languages, for short.[1]

Claim 1 *The 0-bundled languages $\prod^{0\text{-}bnd}_{a\in A} L^a_{pp}$ is the Cartesian product (direct product) of the languages $\prod_{a\in A} L^a_{pp}$.*

Claim 2 *The k-bundled languages $\prod^{k\text{-}bnd}_{a\in A} L^a_{pp}$ is a subset of the Cartesian product of the language $\prod_{a\in A} L^a_{pp}$.*

The both claims are deduced from (8).

5 Decentralized Multi-authority Anonymous Authentication with Non-interactive Proofs

In this section, we give a syntax and security definitions of a decentralized multi-authority anonymous authentication scheme with non-interactive proofs, which we call NI-DMA-A-AUTH for short. In the security definitions we capture the requirements described in Section 1: One is resistance against collusion attacks that cause misauthentication, and the other is anonymity for privacy protection.

5.1 Syntax

Our scheme $\mathsf{a\text{-}auth}$ consists of five PPT algorithms, (Setup, AuthKG_{pp}, SKG_{pp}, Prover_{pp}, $\mathsf{Verifier}_{pp}$).

• $\mathsf{Setup}(1^\lambda) \to pp$. This PPT algorithm is needed to generate a set of public parameters pp. On input the security parameter 1^λ, it generates the set pp. It returns pp.

• $\mathsf{AuthKG}_{pp}(a) \to (\mathrm{PK}^a, \mathrm{MSK}^a)$. This PPT algorithm is executed by a key-issuing authority indexed by a positive integer a. On input the authority index a, it generates the a-th public key PK^a of the authority and the corresponding a-th master secret key MSK^a. It returns $(\mathrm{PK}^a, \mathrm{MSK}^a)$.

• $\mathsf{SKG}_{pp}(\mathrm{PK}^a, \mathrm{MSK}^a, \mathsf{gid}) \to \mathrm{sk}^a_{\mathsf{gid}}$. This PPT algorithm is executed by the a-th key-issuing authority. On input the a-th public and master secret keys $(\mathrm{PK}^a, \mathrm{MSK}^a)$ and an element $\mathsf{gid} \in \check{G}$ of a prover, it generates a private secret key $\mathrm{sk}^a_{\mathsf{gid}}$ of a prover. It returns $\mathrm{sk}^a_{\mathsf{gid}}$.

• $\mathsf{Prover}_{pp}((\mathrm{PK}^a, \mathrm{sk}^a_{\mathsf{gid}})^{a\in A'}) \to \pi$. This PPT algorithm is executed by a prover who is to be authenticated, where A' denotes a subset of all indices at which the prover is issued her private secret keys by authorities. On input the public keys $(\mathrm{PK}^a)^{a\in A'}$ and the corresponding private secret keys $(\mathrm{sk}^a_{\mathsf{gid}})^{a\in A'}$, it returns a proof π.

• $\mathsf{Verifier}_{pp}((\mathrm{PK}^a)^{a\in A'}, \pi) \to d$. This deterministic polynomial-time algorithm is executed by a verifier who confirms that the prover certainly knows the secret keys for indices $a \in A'$. On input the public keys $(\mathrm{PK}^a)^{a\in A'}$ and the proof π, it returns $d := 1$ ("accept") or $d := 0$ ("reject").

[1] Our meaning of "bundled product of languages" is a subset of the Cartesian product (direct product), where the witness spaces exist as a family of a space parametrized by the base point $(w_0, \ldots, w_{k-1})$ like "vector bundle" [AA18a,AA18b]. (We would like to avoid confusion that "bundled product of languages" is recognized as something sold as "product bundle".)

5.2 Security Definitions

We discuss two security notions for our authentication scheme a-auth; security against collusion attacks that yield misauthentication, and anonymity for privacy of provers' global identities.

Resistance against Concurrent and Collusion Attack of Misauthentication. One of the strongest attacks to cause misauthentication is the concurrent and collusion attack on our a-auth. For a formal treatment we define the following experiment on a-auth and an adversary algorithm $\mathbf{A}$.

$$
\begin{aligned}
&\mathsf{Exp}^{\text{conc-coll}}_{\text{a-auth},\mathbf{A}}(1^\lambda):\\
&(q_A, St) \leftarrow \mathbf{A}(1^\lambda), A := \{1,\ldots,q_A\}, pp \leftarrow \mathsf{Setup}(1^\lambda)\\
&\text{For } a \in A : (\mathrm{PK}^a, \mathrm{MSK}^a) \leftarrow \mathsf{AuthKG}_{pp}(a)\\
&((\mathtt{gid}_i)_{i=1}^{q_I}, St) \leftarrow \mathbf{A}(St, pp, (\mathrm{PK}^a)^{a\in A}), I := \{1,\ldots,q_I\}\\
&\text{For } a \in A : \text{For } i \in I : \mathrm{sk}^a_{\mathtt{gid}_i} \leftarrow \mathsf{SKG}_{pp}(\mathrm{PK}^a, \mathrm{MSK}^a, \mathtt{gid}_i)\\
&(A^*, \pi^*) \leftarrow \mathbf{A}^{\mathsf{Prover}_{pp}((\mathrm{PK}^a,\mathrm{sk}^a_{\mathtt{gid}_i})^{a\in A})|_{i\in I}, \mathbf{SKO}_{pp}(\mathrm{PK}^{\cdot},\mathrm{MSK}^{\cdot},\cdot)}(St)\\
&\mathsf{Verifier}_{pp}((\mathrm{PK}^a)^{a\in A^*}, \pi^*) \to d\\
&\text{If } d = 1 \text{ then return } \textsc{Win} \text{ else return } \textsc{Lose}
\end{aligned}
$$

Intuitively, the above experiment describes the attack as follows. The adversary algorithm $\mathbf{A}$, on input the security parameter 1^λ, first outputs the number q_A of key-issuing authorities. Then, on input the set of public parameters pp and the issued public keys $(\mathrm{PK}^a)^{a\in A}$, $\mathbf{A}$ outputs global identity element $\mathtt{gid}_i \in \check{G}, i = 1,\ldots,q_I$. $\mathbf{A}$ invokes prover algorithm Prover_{pp} with $\mathtt{gid}_i, i = 1,\ldots,q_I$ to obtain proofs. In addition, $\mathbf{A}$ collects at most q_{sk} private secret keys by issuing queries to the private secret key oracle $\mathbf{SKO}_{pp}(\mathrm{PK}^{\cdot}, \mathrm{MSK}^{\cdot}, \cdot)$ with an authority index $a \in A$ and a global identity element $\mathtt{gid}_j \in \check{G}$ for $j = q_I + 1,\ldots,q_I + q_{\mathrm{sk}}$. We denote by A_j the set of authority indices for which the queries with $\mathtt{gid}_j$ were issued. That is,

$$A_j := \{a \in A \mid \mathbf{A} \text{ receives } \mathrm{sk}^a_{\mathtt{gid}_j}\}, j = q_I + 1,\ldots,q_I + q_{\mathrm{sk}}.$$

We here require that the numbers q_A, q_I and q_{sk} are bounded by a polynomial in λ. At the end $\mathbf{A}$ returns a target set of authority indices and a forgery proof (A^*, π^*). If the decision d on π^* by $\mathsf{Verifier}_{pp}$ is 1 under $(\mathrm{PK}^a)^{a\in A^*}$, then the experiment returns Win; otherwise it returns Lose.

A restriction is imposed on the adversary $\mathbf{A}$: The target set of authority indices A^* should not be a subset of any single set A_j:

$$A^* \not\subseteq A_j,\ j = q_I + 1,\ldots,q_I + q_{\mathrm{sk}}. \tag{9}$$

This restriction is because otherwise, $\mathbf{A}$ is given private secret keys for A^* on a single $\mathtt{gid}_{j^*}$ for some j^*, $q_I < j^* \le q_I + q_{\mathrm{sk}}$, and then $\mathbf{A}$ can trivially succeed in causing misauthentication.

The advantage of an adversary $\mathbf{A}$ over an authentication scheme a-auth in the experiment is defined as: $\mathbf{Adv}^{\text{conc-coll}}_{\text{a-auth},\mathbf{A}}(\lambda) \stackrel{\text{def}}{=} \Pr[\mathsf{Exp}^{\text{conc-coll}}_{\text{a-auth},\mathbf{A}}(1^\lambda) = \textsc{Win}]$. A scheme a-auth is called secure against concurrent and collusion attacks that cause misauthentication. if, for any PPT algorithm $\mathbf{A}$, the advantage $\mathbf{Adv}^{\text{conc-coll}}_{\text{a-auth},\mathbf{A}}(\lambda)$ is negligible in λ.

Anonymity. As is explained in Section 1, a critical feature to be attained is provers' anonymity on global identities when the provers are authenticated. Formally we define the following experiment on a-auth and an adversary algorithm $\mathbf{A}$.

$$
\begin{aligned}
&\mathsf{Exp}^{\text{ano}}_{\text{a-auth},\mathbf{A}}(1^\lambda):\\
&(q_A, St) \leftarrow \mathbf{A}(1^\lambda), A := \{1,\ldots,q_A\}, pp \leftarrow \mathsf{Setup}(1^\lambda)\\
&\text{For } a \in A : (\mathrm{PK}^a, \mathrm{MSK}^a) \leftarrow \mathsf{AuthKG}_{pp}(1^\lambda, a)\\
&(\mathtt{gid}_0, \mathtt{gid}_1, St) \leftarrow \mathbf{A}(St, pp, (\mathrm{PK}^a)^{a\in A})\\
&\text{For } a \in A : \text{For } i = 0,1 : \mathrm{sk}^a_{\mathtt{gid}_i} \leftarrow \mathsf{SKG}_{pp}(\mathrm{PK}^a, \mathrm{MSK}^a, \mathtt{gid}_i)\\
&b \in_U \{0,1\}, b' \leftarrow \mathbf{A}^{\mathsf{Prover}_{pp}((\mathrm{PK}^a,\mathrm{sk}^a_{\mathtt{gid}_b})^{a\in A})}(St, (\mathrm{sk}^a_{\mathtt{gid}_0}, \mathrm{sk}^a_{\mathtt{gid}_1})^{a\in A})\\
&\text{If } b = b', \text{ then return } \textsc{Win}, \text{ else return } \textsc{Lose}
\end{aligned}
$$

Intuitively, the above experiment describes the attack as follows. The adversary algorithm $\mathbf{A}$, on input the security parameter 1^λ, first outputs the number q_A of key-issuing authorities. Then, on input the set of public parameters pp and the issued public keys $(\mathrm{PK}^a)^{a\in A}$, $\mathbf{A}$ designates two identity elements $\mathtt{gid}_0$ and $\mathtt{gid}_1$. Next, $\mathbf{A}$ interacts with a prover Prover_{pp} on input the private secret keys $(\mathrm{sk}^a_{\mathtt{gid}_b})^{a\in A}$, where the index b is chosen uniformly at random and hidden from $\mathbf{A}$. If the decision b' of $\mathbf{A}$ is equal to b, then the experiment returns Win; otherwise it returns Lose.

The advantage of an adversary $\mathbf{A}$ over an authentication scheme $\mathsf{a\text{-}auth}$ in the experiment is defined as: $\mathbf{Adv}^{\mathrm{ano}}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda) \stackrel{\mathrm{def}}{=} \left|\Pr[\mathsf{Exp}^{\mathrm{ano}}_{\mathsf{a\text{-}auth},\mathbf{A}}(1^\lambda) = \mathrm{Win}] - (1/2)\right|$. An authentication scheme $\mathsf{a\text{-}auth}$ is called to have anonymity if, for any PPT algorithm $\mathbf{A}$, the advantage $\mathbf{Adv}^{\mathrm{ano}}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda)$ is negligible in λ.

6 Construction and Security Proofs of NI-DMA-A-AUTH

In this section, we construct a scheme of NI-DMA-A-AUTH. We employ two building blocks. One is the structure-preserving signature scheme [AFG+10,AFG+16] (see Section 2.2). Each decentralized authority indexed by 'a' issues a private secret key $\mathrm{sk}^a_{\mathtt{gid}}$ for a global identity element $\mathtt{gid}$. The other building block is the non-interactive commit-and-prove scheme of the fine-tuned Groth-Sahai proof system [GS08,EG14] adapted to the case of the structure-preserving signature (see Section 3). In the commit-phase a prover generates commitments to the global identity element $\mathtt{gid}$ and the components of the structure-preserving signatures $(\sigma^a_1,\ldots,\sigma^a_7)^{a\in A'}$. In the prove-phase the prover generates a proof π using the bundled witnesses. That is, $w_0 = \mathtt{gid}$ is the common component, and for $(w^a_1,\ldots,w^a_7) = (\sigma^a_1,\ldots,\sigma^a_7)$, $(w_0, w^a_1,\ldots,w^a_7)$ is a whole witness of the a-th component. The proof π is a proof for the *bundled* languages (see Section 4).

6.1 Construction

According to the syntax in Section 5, the scheme $\mathsf{a\text{-}auth}$ consists of five PPT algorithms: $\mathsf{a\text{-}auth} = (\mathsf{Setup}, \mathsf{AuthKG}_{pp}, \mathsf{SKG}_{pp}, \mathsf{Prover}_{pp}, \mathsf{Verifier}_{pp})$.

- $\mathsf{Setup}(1^\lambda) \to pp$. On input the security parameter 1^λ, it runs the generation algorithm of bilinear groups, and it puts the output as a set of public parameters: $\mathcal{G}(1^\lambda) \to (p, \hat{\mathbb{G}}, \check{\mathbb{H}}, \mathbb{T}, e, \hat{G}, \check{H}) =: pp$. Note that pp is a common for both the structure-preserving signature scheme Sig and the commit-and-prove scheme CmtPrv. Besides, it runs the generation algorithm of commitment key: $\mathsf{Cmt.KG}_{pp}(\mathtt{nor}) \to ck$. It returns $pp := (pp, ck)$.
- $\mathsf{AuthKG}_{pp}(a) \to (\mathrm{PK}^a, \mathrm{MSK}^a)$. On input an authority index a, it executes the key-generation algorithm $\mathsf{Sig.KG}_{pp}()$ to obtain $(\mathrm{PK}, \mathrm{SK})$. It puts $\mathrm{PK}^a := \mathrm{PK}$ and $\mathrm{MSK}^a := \mathrm{SK}$. It returns $(\mathrm{PK}^a, \mathrm{MSK}^a)$.
- $\mathsf{SKG}_{pp}(\mathrm{PK}^a, \mathrm{MSK}^a, \mathtt{gid}) \to \mathrm{sk}^a_{\mathtt{gid}}$. On input PK^a, MSK^a and an element $\mathtt{gid} \in \check{\mathbb{H}}$, it puts $\mathrm{PK}^a := \mathrm{PK}^a$ and $\mathrm{SK}^a := \mathrm{MSK}^a$ and $m := \check{M} := \mathtt{gid}$. It executes the signing algorithm $\mathsf{Sig.Sign}_{pp}(\mathrm{PK}^a, \mathrm{SK}^a, m)$ to obtain a signature σ^a. It puts $\mathrm{sk}^a_{\mathtt{gid}} := (\mathtt{gid}, \sigma^a)$. It returns $\mathrm{sk}^a_{\mathtt{gid}}$.
- $\mathsf{Prover}_{pp}((\mathrm{PK}^a, \mathrm{sk}^a_{\mathtt{gid}})^{a\in A'}) \to \pi$. On input $(\mathrm{PK}^a, \mathrm{sk}^a_{\mathtt{gid}})^{a\in A'}$, first, it commits to $\mathtt{gid}$:

$$c_0 \leftarrow \mathsf{Cmt.Com}_{pp}(\mathtt{gid}; r_0).$$

Second, for each $a \in A'$, it commits to the components $\sigma^a_1,\ldots,\sigma^a_7$ of the signature σ^a in the componentwise way.

$$(c^a_1,\ldots,c^a_7) \leftarrow \mathsf{Cmt.Com}_{pp}((\sigma^a_1,\ldots,\sigma^a_7); (r^a_1,\ldots,r^a_7)).$$

Then, for each $a \in A'$, it puts $x^a := (\hat{G}^a_z, \hat{G}^a_{u,z}, \hat{G}^a_u, \hat{G}^a_1, \hat{G}^a_{u,1}, (\hat{A}^a_i, (\check{A}^a_i)^{-1}, \hat{B}^a_i, (\check{B}^a_i)^{-1})^1_{i=0})$ by using PK^a. It also puts $c^a := (c_0, c^a_1,\ldots,c^a_7)$, $w^a := (w_0, w^a_1,\ldots,w^a_7) := (\mathtt{gid}, \sigma^a_1,\ldots,\sigma^a_7)$ and $r^a := (r_0, r^a_1,\ldots,r^a_7)$. It executes the prove-algorithm to obtain a proof:

$$\pi^a \leftarrow \mathsf{P}_{pp}(x^a, c^a, w^a, r^a), a \in A'.$$

It puts $\bar{\pi}^a := ((c^a_1,\ldots,c^a_7), \pi^a)$ for each $a \in A'$, and it merges all the $\bar{\pi}^a$s and the single commitment c_0 to $\mathtt{gid}$: $\pi := (c_0, (\bar{\pi}^a)^{a\in A'})$. It returns π.

• $\mathsf{Verifier}_{pp}((\mathrm{PK}^a)^{a\in A'}, \pi) \rightarrow d$. On input $((\mathrm{PK}^a)^{a\in A'}, \pi)$, it converts PK^a into x^a and it puts $c^a := (c_0, c_1^a, \ldots, c_7^a)$ for each $a \in A'$. Then it executes the verify-algorithm for each $a \in A'$ to obtain the decisions:

$$d^a \leftarrow \mathsf{V}_{pp}(x^a, c^a, \pi^a), a \in A'.$$

If all the decisions d^as are 1, then it returns $d := 1$; otherwise it returns $d := 0$.

6.2 Security Proofs

Theorem 1 (Resistance against Concurrent and Collusion Attacks) *For any* PPT *algorithm* $\mathbf{A}$ *that is in accordance with the experiment* $\mathsf{Exp}^{conc\text{-}coll}_{a\text{-}auth,\mathbf{A}}(1^\lambda)$*, there exists a* PPT *algorithm* $\mathbf{F}$ *that is in accordance with the experiment* $\mathsf{Exp}^{euf\text{-}cma}_{Sig,\mathbf{F}}(1^\lambda)$ *and the following inequality holds.*

$$\mathbf{Adv}^{conc\text{-}coll}_{a\text{-}auth,\mathbf{A}}(\lambda) < \frac{1}{1-(q_I/2^{\lambda-1})-(q_{sk}/2^{\lambda-1})} \cdot q_A \cdot \mathbf{Adv}^{euf\text{-}cma}_{Sig,\mathbf{F}}(\lambda).$$

This theorem means that, if the structure-preserving signature scheme Sig is existentially unforgeable against adaptive chosen-message attacks, then our a-auth is secure against concurrent and collusion attacks.

Proof. Given any PPT algorithm $\mathbf{A}$ that is in accordance with the experiment $\mathsf{Exp}^{\text{conc-coll}}_{\mathsf{a\text{-}auth},\mathbf{A}}(1^\lambda)$, we construct a PPT algorithm $\mathbf{F}$ that generates an existential forgery of Sig in accordance with the experiment $\mathsf{Exp}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(1^\lambda)$. $\mathbf{F}$ is given as input the set of public parameters pp and a public key PK. $\mathbf{F}$ executes $\mathsf{Cmt.KG}_{pp}(\mathtt{ext})$ to obtain a pair (ck, xk). $\mathbf{F}$ puts $pp := (pp, ck)$. $\mathbf{F}$ invokes the algorithm $\mathbf{A}$ with 1^λ to obtain the number q_A of authorities. $\mathbf{F}$ chooses a *target index* a^* from the set $A := \{1, \ldots, q_A\}$ uniformly at random. $\mathbf{F}$ runs the authority key generation algorithm honestly for $a \in A$ *except* the target index a^*. As for a^*, $\mathbf{F}$ uses the input public key:

$$\begin{aligned}&\text{For } a \in A \text{ s.t. } a \neq a^* : (\mathrm{PK}^a, \mathrm{MSK}^a) \leftarrow \mathsf{AuthKG}_{pp}(a),\\ &\text{For } a = a^* : \mathrm{PK}^{a^*} := \mathrm{PK}.\end{aligned}$$

$\mathbf{F}$ inputs pp and $(\mathrm{PK}^a)^{a\in A}$ into $\mathbf{A}$ to obtain the number q_I of concurrent provers. $\mathbf{F}$ puts $I := \{1, \ldots, q_I\}$.
Simulation of Concurrent Provers. When $\mathbf{A}$ invokes a prover with $\mathtt{gid}_i \in \check{G}$, $i = 1, \ldots, q_I$, $\mathbf{F}$ runs the generation algorithm of a private secret key with $\mathtt{gid}_i$ honestly for $a \in A$ *except* the target index a^*. As for a^*, $\mathbf{F}$ issues a signing query with $\mathtt{gid}_i$ to its oracle:

$$\begin{aligned}&\text{For } a \in A \text{ s.t. } a \neq a^* : \mathrm{sk}^a_{\mathtt{gid}_i} \leftarrow \mathsf{SKG}_{pp}(\mathrm{PK}^a, \mathrm{MSK}^a, \mathtt{gid}_i),\\ &\text{For } a = a^*, \mathrm{sk}^{a^*}_{\mathtt{gid}_i} \leftarrow \mathbf{SignO}_{pp}(\mathrm{PK}, \mathrm{SK}, \mathtt{gid}_i).\end{aligned}$$

In the simulation of concurrent provers $\mathsf{Prover}_{pp}((\mathrm{PK}^a, \mathrm{sk}^a_{\mathtt{gid}_i})^{a\in A})|_{i\in I}$, $\mathbf{F}$ uses the private secret keys $(\mathrm{sk}^a_{\mathtt{gid}_i})^{a\in A}, i \in I$. Note that this is a perfect simulation.
Simulation of Private Secret Key Oracle. When $\mathbf{A}$ issues a private secret key query with $A_j \subset A$ and $\mathtt{gid}_j \in \check{G}$, $j = q_I + 1, \ldots, q_I + q_{\mathrm{sk}}$, $\mathbf{F}$ runs the generation algorithm of private secret key with $\mathtt{gid}_j$ honestly for $a \in A$ *except* the target index a^*. As for a^*, $\mathbf{F}$ issues a signing query with $\mathtt{gid}_j$ to its oracle:

$$\begin{aligned}&\text{For } a \in A \text{ s.t. } a \neq a^* : \mathrm{sk}^a_{\mathtt{gid}_j} \leftarrow \mathsf{SKG}_{pp}(\mathrm{PK}^a, \mathrm{MSK}^a, \mathtt{gid}_j),\\ &\text{For } a = a^* : \mathrm{sk}^{a^*}_{\mathtt{gid}_j} \leftarrow \mathbf{SignO}_{pp}(\mathrm{PK}, \mathrm{SK}, \mathtt{gid}_j).\end{aligned}$$

$\mathbf{F}$ replies to $\mathbf{A}$ with the secret key $\mathrm{sk}^a_{\mathtt{gid}_j}$. This is also a perfect simulation.

At the end $\mathbf{A}$ returns a target set of authority indices and a forgery proof (A^*, π^*).
Generating Existential Forgery. Next, $\mathbf{F}$ runs a verifier $\mathsf{Verifier}_{pp}$ with an input $((\mathrm{PK}^a)^{a\in A^*}, \pi^*)$. If the decision d of $\mathsf{Verifier}_{pp}$ is 1, then $\mathbf{F}$ executes for each $a \in A^*$ the extraction algorithm $\mathsf{Cmt.Ext}_{pp}(xk, c^a)$ to obtain a committed message $(w^a)^* = ((w_0^a)^*, (w_1^a)^*, \ldots, (w_7^a)^*)$. Note here that, for all $a \in A^*$, $(w_0^a)^*$ is equal to a single element $(w_0)^*$ in $\check{G}$. *This is because* of the perfectly binding property of Cmt_{pp}. Then $\mathbf{F}$ puts $\mathtt{gid}^* := (w_0)^*$. The restriction (9) of the experiment assures that there exists at least one authority index $\hat{a}$ such that $\hat{a} \in A^*$ and $\hat{a} \notin A_j$ for $j = q_I + 1, \ldots, q_I + q_{\mathrm{sk}}$. $\mathbf{F}$ chooses such an $\hat{a}$ at random and puts

$\sigma^* := (\sigma^{\hat{a}})^* := ((w_1^{\hat{a}})^*, \ldots, (w_7^{\hat{a}})^*)$. $\mathbf{F}$ returns a forgery pair of a message and a signature $(\mathsf{gid}^*, \sigma^*)$. This completes the description of $\mathbf{F}$.

Probability Evaluation. The probability that the returned value $(\mathsf{gid}^*, \sigma^*)$ is actually an existential forgery is evaluated as follows. We name the events in the above $\mathbf{F}$ as:

$$\begin{aligned}
\textsc{Acc} &: \mathsf{Verifier}_{pp} \text{ accepts } \mathbf{A},\\
\textsc{TgtIdx} &: \hat{a} = a^*,\\
\textsc{Ext} &: \mathsf{Cmt.Ext}_{pp} \text{ returns a witness } (w^a)^*\\
\textsc{NewID} &: \mathsf{gid}^* \notin \{\mathsf{gid}_i\}_{i=1}^{q_I+q_{\mathrm{sk}}},\\
\textsc{Forge} &: (\mathsf{gid}^*, \sigma^*) \text{ is an existential forgery on } \mathsf{Sig}.
\end{aligned}$$

We have the following equalities.

$$\mathbf{Adv}^{\text{conc-coll}}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda) = \Pr[\textsc{Acc}], \tag{10}$$

$$\Pr[\textsc{Acc}, \textsc{Ext}, \textsc{TgtIdx}, \textsc{NewID}] = \Pr[\textsc{Forge}], \tag{11}$$

$$\Pr[\textsc{Forge}] = \mathbf{Adv}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(\lambda). \tag{12}$$

The left-hand side of the equality (11) is expanded as follows.

$$\begin{aligned}
&\Pr[\textsc{Acc}, \textsc{Ext}, \textsc{TgtIdx}, \textsc{NewID}]\\
=&\Pr[\textsc{TgtIdx}] \cdot \Pr[\textsc{Acc}, \textsc{Ext}, \textsc{NewID}]\\
=&\Pr[\textsc{TgtIdx}] \cdot \Pr[\textsc{Acc}, \textsc{Ext}] \cdot \Pr[\textsc{NewID} \mid \textsc{Acc}, \textsc{Ext}]\\
=&\Pr[\textsc{TgtIdx}] \cdot \Pr[\textsc{Acc}] \cdot \Pr[\textsc{Ext} \mid \textsc{Acc}] \cdot \Pr[\textsc{NewID} \mid \textsc{Acc}, \textsc{Ext}].
\end{aligned} \tag{13}$$

Claim 3

$$\Pr[\textsc{TgtIdx}] = 1/q_A. \tag{14}$$

Proof. The restriction (9) assures that there exists an authority index $\hat{a}$ such that $\hat{a} \in A^*$ and $\hat{a} \notin A_j, q_I+1 \leq \forall j \leq q_I + q_{\mathrm{sk}}$. $\hat{a}$ coincides with a^* with probability $1/q_A$. □

Claim 4

$$\Pr[\textsc{NewID} \mid \textsc{Acc}, \textsc{Ext}] \geq \frac{p - q_I - q_{sk}}{p}. \tag{15}$$

Proof. gid^* is not in $\{\mathsf{gid}_i\}_{i=1}^{q_I+q_{\mathrm{sk}}}$ with probability at least $\frac{p-q_I-q_{\mathrm{sk}}}{p}$. □

Claim 5

$$\Pr[\textsc{Ext} \mid \textsc{Acc}] = 1. \tag{16}$$

Proof. This is because of the perfect knowledge extraction of Π_{pp}. □

Combining (10), (11), (12), (13), (14), (15) and (16) we have:

$$\mathbf{Adv}^{\text{conc-coll}}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda) \leq \frac{p}{p - q_I - q_{\mathrm{sk}}} \cdot q_A \cdot \mathbf{Adv}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(\lambda). \tag{17}$$

Here p is a prime number of bit-length λ, so $p > 2^{\lambda-1}$. Then we have:

$$\mathbf{Adv}^{\text{conc-coll}}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda) < \frac{1}{1 - (q_I/2^{\lambda-1}) - (q_{\mathrm{sk}}/2^{\lambda-1})} \cdot q_A \cdot \mathbf{Adv}^{\text{euf-cma}}_{\mathsf{Sig},\mathbf{F}}(\lambda). \tag{18}$$

□

Theorem 2 (Anonymity) *Assume the computational indistinguishability between commitment keys $\{ck\}$ of the mode* `nor` *and commitment keys $\{ck\}$ of the mode* `sim`*. For any* PPT *algorithm* $\mathbf{A}$ *that is in accordance with the experiment $\mathsf{Exp}^{ano}_{\mathsf{a\text{-}auth},\mathbf{A}}(1^\lambda)$, there exists a* PPT *algorithm* $\mathbf{D}$ *and the following equality holds.*

$$\mathbf{Adv}^{ano}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda) \leq \mathbf{Adv}^{ind\text{-}dual}_{Cmt_{pp},\mathbf{D}}(\lambda).$$

This theorem means that, if the dual-mode commitment keys are indistinguishable, then our a-auth has anonymity.

Proof. Suppose that any PPT algorithm $\mathbf{A}$ that is in accordance with the experiment $\mathsf{Exp}^{\mathrm{ano}}_{\mathsf{a\text{-}auth},\mathbf{A}}(1^\lambda)$ is given. We set a sequence of games, Game_0 and Game_1, as follows. Game_0 is exactly the same as $\mathsf{Exp}^{\mathrm{ano}}_{\mathsf{a\text{-}auth},\mathbf{A}}(1^\lambda)$. Note that when a set of public parameters $pp = (pp', ck)$ is given to $\mathbf{A}$ where pp' is for bilinear groups, the commitment key ck is chosen as a commitment key ck of the mode `nor`. We denote the probability that Game_0 returns Win as $\Pr[\mathrm{Win}_0]$.

Game_1 is Game_0 except that, when a set of public parameters $pp = (pp', ck)$ is given to $\mathbf{A}$, the commitment key ck is chosen as a commitment key ck of the mode `sim`. We denote the probability that Game_1 returns Win as $\Pr[\mathrm{Win}_1]$. The values in Game_1 distribute identically for both gid_0 and gid_1 due to the perfectly hiding property (6) and the witness-indistinguishability (7). Therefore, $\Pr[\mathrm{Win}_1] = 1/2$.

Employing $\mathbf{A}$ as a subroutine, we construct a PPT distinguisher algorithm $\mathbf{D}$ as follows. Given an input pp, ck, $\mathbf{D}$ reads out the security parameter. $\mathbf{D}$ simulates the environment of $\mathbf{A}$ in Game_0 or Game_1 honestly except that $\mathbf{D}$ puts $pp := (pp, ck)$ instead of executing $\mathsf{Setup}(1^\lambda)$. If $b = b'$, then $\mathbf{D}$ returns 1, and otherwise, 0. By the definition of (5), $\Pr[\mathbf{D}(pp, ck) = 1 \mid ck \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{nor})] = \Pr[\mathrm{Win}_0]$ and $\Pr[\mathbf{D}(pp, ck) = 1 \mid (ck, tk) \leftarrow \mathsf{Cmt.KG}_{pp}(\mathtt{sim})] = \Pr[\mathrm{Win}_1]$, and

$$\mathbf{Adv}^{\text{ind-dual}}_{\mathsf{Cmt}_{pp},\mathbf{D}}(\lambda) = |\Pr[\mathrm{Win}_0] - \Pr[\mathrm{Win}_1]|. \tag{19}$$

Therefore,

$$\begin{aligned}\mathbf{Adv}^{\text{ano}}_{\mathsf{a\text{-}auth},\mathbf{A}}(\lambda) &= |\Pr[\mathrm{Win}_0] - (1/2)| \\ &\leq |\Pr[\mathrm{Win}_0] - \Pr[\mathrm{Win}_1]| + |\Pr[\mathrm{Win}_1] - (1/2)| \\ &= \mathbf{Adv}^{\text{ind-dual}}_{\mathsf{Cmt}_{pp},\mathbf{D}}(\lambda) + 0 = \mathbf{Adv}^{\text{ind-dual}}_{\mathsf{Cmt}_{pp},\mathbf{D}}(\lambda). \end{aligned} \tag{20}$$

□

7 Conclusion

We gave a NI-DMA-A-AUTH scheme a-auth, in which a prover is able to convince a verifier that a single anonymous prover has the knowledge of plural attribute credentials issued by independent authorities. Perfect binding property of the commitment to gid works as a proof of simultaneous satisfiability of the verification equations of structure-preserving signatures. Hence the collusion attacks are prevented. On the other hand, perfectly hiding property of commitments and perfect witness-indistinguishable property of proofs of the Groth-Sahai proof system yields anonymity, and hence, assures privacy. Other structure-preserving signature schemes such as [AHN+17] can be employed instead of that of [AFG+10,AFG+16].

References

[AA18a] Hiroaki Anada and Seiko Arita. Witness-indistinguishable arguments with Σ-protocols for bundled witness spaces and its application to global identities. In *Information and Communications Security - 20th International Conference, ICICS 2018, Lille, France, October 29-31, 2018, Proceedings*, pages 530–547, 2018.

[AA18b] Hiroaki Anada and Seiko Arita. Witness-indistinguishable arguments with Σ-protocols for bundled witness spaces and its application to global identities. *IACR Cryptology ePrint Archive*, 2018/742, 2018.

[AFG+10] Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Structure-preserving signatures and commitments to group elements. In *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, pages 209–236, 2010.

[AFG+16] Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Structure-preserving signatures and commitments to group elements. *Journal of Cryptology*, 29(2):363–421, 2016.

[AHN+17] Masayuki Abe, Dennis Hofheinz, Ryo Nishimaki, Miyako Ohkubo, and Jiaxin Pan. Compact structure-preserving signatures with almost tight security. In *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II*, pages 548–580, 2017.

[CDHK15] Jan Camenisch, Maria Dubovitskaya, Kristiyan Haralambiev, and Markulf Kohlweiss. Composable and modular anonymous credentials: Definitions and practical constructions. In *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, pages 262–288, 2015.

[CDT19] Jan Camenisch, Manu Drijvers, and Björn Tackmann. Multi-protocol UC and its use for building modular and efficient protocols. *IACR Cryptology ePrint Archive*, 2019:65, 2019.

[CL01] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *Advances in Cryptology - EUROCRYPT 2001, International Conference on the Theory and Application of Cryptographic Techniques, Innsbruck, Austria, May 6-10, 2001, Proceeding*, pages 93–118, 2001.

[CL02] Jan Camenisch and Anna Lysyanskaya. Dynamic accumulators and application to efficient revocation of anonymous credentials. In *Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings*, pages 61–76, 2002.

[Dam10] Ivan Damgård. On σ-protocols. In Course Notes, http://cs.au.dk/ ivan/CPT.html, 2010.

[EG14] Alex Escala and Jens Groth. Fine-tuning Groth-Sahai proofs. In *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 630–649, 2014.

[GPS08] Steven D. Galbraith, Kenneth G. Paterson, and Nigel P. Smart. Pairings for cryptographers. *Discrete Applied Mathematics*, 156(16):3113–3121, 2008.

[GS08] Jens Groth and Amit Sahai. Efficient non-interactive proof systems for bilinear groups. In *Proceedings of the Theory and Applications of Cryptographic Techniques 27th Annual International Conference on Advances in Cryptology*, EUROCRYPT'08, pages 415–432, Berlin, Heidelberg, 2008. Springer-Verlag.

[NIS13] NIST. Digital signature standard (DSS), July 2013.

[OT13] Tatsuaki Okamoto and Katsuyuki Takashima. Decentralized attribute-based signatures. In *Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26 - March 1, 2013. Proceedings*, pages 125–142, 2013.

[SNBF17] Shahidatul Sadiah, Toru Nakanishi, Nasima Begum, and Nobuo Funabiki. Accumulator for monotone formulas and its application to anonymous credential system. *JIP*, 25:949–961, 2017.