Anonymity-enhanced Pseudonym System

Yuko Tamura and Atsuko Miyaji

1-1, Asahidai, Tatsunokuchi, Ishikawa, 923-1292, Japan
{yuko, miyaji}@jaist.ac.jp

Abstract. Pseudonym systems allow users to interact with multiple organizations anonymously by using pseudonyms. Such schemes are of significant practical relevance because it is the best means of providing privacy for users. In previous works, users transact with a organization by demonstration of possession of a credential issued by the organization or relationship with another credential. However, the information that a user has a credential from a specific organization compromises privacy of the user. In the present paper, we give a formal definition of practical pseudonym system in which the level of privacy provided can be chosen be according to security policies.

1 Introduction

As information gets increasingly accessible, it has been important that individuals control their information to protect their privacy. Pseudonym systems (also called anonymous credential systems) [1–6] allow users to work effectively and anonymously with multiple organizations by using different *pseudonyms*. Such systems are called anonymous when transactions carried out by the same user cannot be correlated. In the systems, an organization knows users by only pseudonym in which each pseudonym cannot be linked to others. An organization issues a *credential* on a pseudonym, and the corresponding user demonstrates the possession of this credential to another organization without revealing anything but the possession.

Lysyanskaya, Rivest, Sahai and Wolf [5] proposed a general pseudonym system based on one-way functions and general zero-knowledge proofs. In their scheme, however, credentials for a user need to be reissued by the organization so that the user can prove the possession of a credential several times. Camenisch and Lysyanskaya [6] solved such a problem by applying strong-RSA-based signature schemes [14] and group signature schemes [9] to their pseudonym system. In their scheme, users can demonstrate the possession of credentials in any number of times and these demonstrations cannot be linked to the same pseudonym.

However, unfortunately, the previous schemes [1–6], proving the possession of a credential cannot but give a verifier the information about which organization a user transacts with. A pseudonym system by Camenisch and Lysyanskaya [6] is the most efficient and the practical in previous works. In their system, a user establishes a pseudonym and its validating tag, then the user is issued a credential as a signature on the tag by the organization. Their system requires

the public-key of an issuing organization to prove the possession of a credential and thus gives the information of the organization to a verifier necessarily. As a result, this compromises the privacy of users, although the verifier does not necessarily need the information. The best pseudonym system should be able to choose the level of privacy according to its security policies.

In this paper, we propose an anonymity-enhanced pseudonym system by showing a credential issued by a group. Our system can allow a user to choose the level of privacy according to their security policies. In our system, an organization is a member of a group, a credential on a pseudonym with an organization is issued by the group manager. Such a credential is a signature on the validating tag and the public-key of the organization by the group manager. Consequently, the user can prove the possession of a credential from some organization in the group without informing of the organization. Moreover, our system provides a feature: *flexibility of choosing the methods to prove the possession of a credential* according to security policies. Namely, a user is given the four methods to prove: (1) showing a credential *with* the identity of an organization if a user needs to inform a verifier of the possession of a credential from the organization, (2) showing a credential *without* the identity of an organization if a user wants to give a verifier only the information that the credential is issued from a group, (3) transferring a credential *with* the identity of an organization and (4) transferring a credential *without* the identity of an organization if a user want to prove the possession of a credential to another organization with whom the user has established a pseudonym. Furthermore, our system satisfies all the desirable properties that the previous schemes [1–6] have.

The rest of this paper is organized as follows. The next section presents the formal definitions and the requirements of an anonymity-enhanced pseudonym system. In section 3, we propose a practical pseudonym system, after an overview. The security is discussed in section 4.

2 Formal Definitions and Requirements

2.1 The model of pseudonym system

Our pseudonym system is constituted by the following players:

Certification authority (CA) : only entity that knows the user's identity.
Group $(\boldsymbol{G}_I)$: set of organizations.
Group manager (G_I) : only entity that has a secret-key of $\boldsymbol{G}_I$ and grants a credential to a user. (We use CA and $G_0 \in \boldsymbol{G}_0$ as interchangeable name.)
Organization (O_i) : entity which belongs to groups.
User (U) : entity who registers with a group and transacts with an organization in the group by the pseudonym.
Verifier (V) : entity that verifies credentials of users.

Pseudonym systems should satisfy the following properties:

Anonymity of users : Verifiers (Verifying organizations) cannot find out anything about a user, except the fact of the user's ownership of a credentials, even if it cooperates with others.

Unlinkability of pseudonyms : Different pseudonyms of the same user are not linked, even if a group manager or an organization cooperates with others.

Unforgeability of credentials : It is impossible to forge a credential issued by a group manager, even if users, other group managers and organizations team up.

2.2 Ideal credential system

We define an ideal pseudonyms system [6] that relies on a trusted party T as an intermediator that is responsible for the necessary properties of the system. All transactions are made via T. T also ensures anonymity of the users towards the group managers, organizations, and verifiers. For an ideal pseudonym system (IPS) and a cryptographic pseudonym system without T (CPS), we gives a security definition, same as [6].

Definition 1 *Let $V = poly(k)$ be the number of players in the system with security parameter k. For an ideal pseudonym system IPS, and its cryptographic implementation CPS, we denote a credential system with security parameter k and event scheduler E for the events that take place in this system, by $IPS(1^k, E)$ (resp., $(CPS(1^k, E))$). If $\{A_1(1^k), \ldots, A_V(1^k)\}$ is a list of the players's outputs, then we denote these player's outputs by $\{A_1(1^k), \ldots, A_V(1^k)\}^{PS(1^k,E)}$ when all of them, together, exist within a pseudonyms system PS. CPS is secure if there exists a simulator $\mathcal{S}$ (ideal-world adversary) such that the following holds, for all interactive probabilistic polynomial-time machines $\mathcal{A}$ (real-world adversary), for all sufficiently large k:*
(1) In the IPS, $\mathcal{S}$ controls the players in the ideal-world corresponding to those real world players controlled by $\mathcal{A}$.
(2) For all event schedulers $E^{\mathcal{A}}$

$$\{\{Z_i(1^k)\}_{i=1}^{V}, \mathcal{A}(1^k)\}^{CPS(1^k,E)} \stackrel{c}{\approx} \{\{Z_i(1^k)\}_{i=1}^{V}, \mathcal{S}^{\mathcal{A}}(1^k)\}^{IPS(1^k,E)}$$

where $\mathcal{S}$ is given black-box access to $\mathcal{A}$, ($D_1(1^k) \stackrel{c}{\approx} D_2(1^k)$ denotes computational indistinguishability of the distributions D_1 and D_2.)

2.3 Functional definitions

This section provides functional definitions in our pseudonym system. Let k be the security parameter and $neg(k)$ denote any function that vanishes faster than any inverse polynomial.

Definition 2 *A pseudonym system consists of the following procedure:*

Key generation– GK_G, $GK_{(O,G)}$ *and* GK_U *for* $G, O \in \boldsymbol{G}$ *and* U *output a secret and public-key pair* $(\mathcal{X}_G, \mathcal{Y}_G)$ *for a group* $\boldsymbol{G}$, $(\mathcal{X}_{(O,G)}, \mathcal{Y}_{(O,G)})$ *for an organization* $O \in \boldsymbol{G}$, *and* $(\mathcal{X}_U, \mathcal{Y}_U)$ *for a user* U, *respectively.* GK_G *and* GK_U *take as input* 1^k, *and* $GK_{(U,O)}$ *takes* 1^k *and a group public-key* $\mathcal{Y}_G$.

Pseudonym generation– $GP\langle U, X\rangle$ *between* U *and an entity* $X \in \boldsymbol{G}$, *takes as* U*'s private input the secret-key* $\mathcal{X}_U$, *and as their common input a group public-key* $\mathcal{Y}_G$. *The private output for* U *is some secret information* $\mathcal{S}_{(U,X)}$, *and the common output is* U*'s pseudonym* $\mathcal{P}_{(U,X)}$.

Credential issue– $IC\langle U, G\rangle$ *between* U *and* $G \in \boldsymbol{G}$, *outputs a credential* $\mathcal{C}_{(U,G)}$ *on* $\mathcal{P}_{(U,G)} \in GP\langle U, G\rangle$. U*'s private input is* $\mathcal{X}_U$ *and* $\mathcal{S}_{(U,G)}$, G*'s private input is a group secret-key* $\mathcal{X}_G$, *and their common input is* $\mathcal{Y}_G$ *and* $\mathcal{P}_{(U,G)}$. ($GP \ni \mathcal{P}$ *means that* GP *outputs* $\mathcal{P}$.)

Pseudonym's validity generation– $GV\langle U, O_i\rangle$ *between* U *and* $O_i \in \boldsymbol{G}$, *outputs a signature on* $\mathcal{P}_{(U,O_i)} \in GP\langle U, O_i\rangle$. O_i*'s private input is a secret-key* $\mathcal{X}_{(O_i,G)}$, *and their common input is* $\mathcal{Y}_G, \mathcal{Y}_{(O_i,G)}$ *and* $\mathcal{P}_{(U,O_i)}$ *with* O_i. U*'s private output is a signature* $\sigma_{(U,O_i)}$.

Credential blind issue– $BIC\langle U, G\rangle$, *blind issue of a credential on a pseudonym, between* U *and* $G \in \boldsymbol{G}$, *outputs a credential* $\mathcal{C}_{(U,O_i)}$ *on* $\mathcal{P}_{(U,O_i)} \in GP\langle U, O_i\rangle$ *where* $O_i \in \boldsymbol{G}$. U*'s private input is* $\mathcal{X}_U, \mathcal{S}_{(U,G)}, \mathcal{S}_{(U,O_i)}$ *and* $\mathcal{P}_{(U,O_i)}$, G*'s private input is* $\mathcal{X}_G$, *and their common input is* $\mathcal{Y}_G, \mathcal{Y}_{(O_i,G)}, \mathcal{P}_{(U,G)}$ *and* $\sigma_{(U,O_i)}$.

Credential showing– $SC\langle U, V\rangle$, *showing a credential on a pseudonym with a group, between* U *and* V, *takes as* U*'s private input* $\mathcal{X}_U, \mathcal{S}_{(U,G)}, \mathcal{P}_{(U,G)}$ *and* $\mathcal{C}_{(U,G)}$, *and as their common input* $\mathcal{Y}_G$. *It outputs* 1 *or* 0, *which, if* $\mathcal{C}_{(U,G)} \in IC\langle U, G\rangle(\mathcal{P}_{(U,G)})$ *where* $\mathcal{P}_{(U,G)} \in GP\langle U, G\rangle$ *or not with probability* $1 - neg(k)$, *respectively.* ($IC(\mathcal{P}) \ni \mathcal{C}$ *means that* IC *outputs* $\mathcal{C}$ *by an input* $\mathcal{P}$.)

$SC^+\langle U, V\rangle$, *showing a credential with identity of an organization, between* U *and* V, *takes as* U*'s private input* $\mathcal{X}_U, \mathcal{S}_{(U,O_i)}, \mathcal{P}_{(U,O_i)}$ *and* $\mathcal{C}_{(U,O_i)}$, *and as their common input* $\mathcal{Y}_G$ *and* $\mathcal{Y}_{(O_i,G)}$. *It outputs* 1 *or* 0, *which, if* $\mathcal{C}_{(U,O_i)} \in BIC\langle U(\mathcal{P}_{(U,O_i)}), G\rangle$ *where* $\mathcal{P}_{(U,O_i)} \in GP\langle U, O_i\rangle$, *or not with probability* $1 - neg(k)$, *respectively.* ($BIC\langle U(\mathcal{P}), G\rangle) \ni \mathcal{C}$ *means that* BIC *outputs* $\mathcal{C}$ *by* U*'s private input* $\mathcal{P}$.)

$SC^-\langle U, V\rangle$, *showing a credential without identity of an organization, between* U *and* V, *takes as* U*'s private input* $\mathcal{X}_U, \mathcal{S}_{(U,O_i)}, \mathcal{P}_{(U,O_i)}, \mathcal{C}_{(U,O_i)}$ *and* $\mathcal{Y}_{(O_i,G)}$, *and as their common input* $\mathcal{Y}_G$. *It outputs* 1 *or* 0, *which, if* $\mathcal{C}_{(U,O_i)} \in BIC\langle U(\mathcal{P}_{(U,O_i)}), G\rangle$ *where* $\mathcal{P}_{(U,O_i)} \in GP\langle U, O_i\rangle$, *or not with probability* $1 - neg(k)$, *respectively.*

Credential transfer– $TC\langle U, X_j\rangle$, *transferring a credential on a pseudonym with a group, between a user* U *and an entity* $X_j \in \boldsymbol{G}_J$, *takes as* U*'s private input* $\mathcal{X}_U, \mathcal{S}_{(U,G_I)}, \mathcal{S}_{(U,X_j)}, \mathcal{P}_{(U,G_I)}$ *and* $\mathcal{C}_{(U,G_I)}$, *as their common input* $\mathcal{Y}_{G_I}, \mathcal{Y}_{G_J}$ *and* $\mathcal{P}_{(U,X_j)}$. *It outputs* 1 *or* 0, *which, if* $\mathcal{C}_{(U,G_I)} \in IC\langle U, G_I\rangle(\mathcal{P}_{(U,G_I)})$, $\mathcal{P}_{(U,G_I)} \in GP\langle U(\mathcal{X}_U), G_I\rangle$ *and* $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$ *or not with probability* $1 - neg(k)$, *respectively.*

$TC^+\langle U, X_j\rangle$, *transferring a credential with identity of an organization, between* U *and* $X_j \in \boldsymbol{G}_J$, *takes as* U*'s private input* $\mathcal{X}_U, \mathcal{S}_{(U,O_i)}$,
$\mathcal{S}_{(U,X_j)}, \mathcal{P}_{(U,O_i)}$ *and* $\mathcal{C}_{(U,O_i)}$, *as their common input* $\mathcal{Y}_{G_I}, \mathcal{Y}_{G_J}, \mathcal{Y}_{(O_i,G_I)}$ *and* $\mathcal{P}_{(U,X_j)}$. *It outputs* 1 *or* 0, *which, if* $\mathcal{C}_{(U,O_i)} \in BIC\langle U(\mathcal{P}_{(U,O_i)}), G_I\rangle$, $\mathcal{P}_{(U,O_i)} \in$

$GP\langle U(\mathcal{X}_U), O_i\rangle$, *and* $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$ *or not with probability* $1 - neg(k)$, *respectively.*
$TC^-\langle U, X_j\rangle$, *transferring a credential without identity of an organization, between* U *and an entity* $X_j \in \boldsymbol{G}_J$, *takes as* U*'s private input* $\mathcal{X}_U, \mathcal{S}_{(U,O_i)}$, $\mathcal{S}_{(U,X_j)}, \mathcal{P}_{(U,O_i)}, \mathcal{C}_{(U,O_i)}$ *and* $\mathcal{Y}_{(O_i,G_I)}$, *and as their common input* $\mathcal{Y}_{G_I}, \mathcal{Y}_{G_J}$ *and* $\mathcal{P}_{(U,X_j)}$. *It outputs* 1 *or* 0, *which, if* $\mathcal{C}_{(U,O_i)} \in BIC\langle U(\mathcal{P}_{(U,O_i)}), G_I\rangle$, $\mathcal{P}_{(U,O_i)} \in GP\langle U(\mathcal{X}_U), O_i\rangle$, *and* $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$ *or not with probability* $1 - neg(k)$, *respectively.*

2.4 Notations

We use the same notation in [6, 9] for the various proofs of knowledge of discrete logarithms and proofs of the validity of statements about discrete logarithms.

(I) *Proof of knowledge or equality in different groups*: We use proofs that the discrete logarithms of two group elements $y_1 \in G_1, y_2 \in G_2$ to the bases $g_1 \in G_1$ and $g_2 \in G_2$ in different groups G_1 and G_2 which has an order q_1 and q_2, respectively, are equal . This proof can be realized only if both discrete logarithms lie in the interval $[0, min\{q_1, q_2\}]$. $PK\{(\alpha) : y_1 = {g_1}^\alpha \wedge y_2 = {g_2}^\alpha \wedge \alpha \in [0, min\{q_1, q_2\}]\}$ denotes a "*zero-knowledge proof of knowledge of integers* α *such that* $y_1 = {g_1}^\alpha$ *and* $y_2 = {g_2}^\alpha$ *holds, where* $\alpha \in [0, min\{q_1, q_2\}]$". This protocol generalized to several different groups, to representations, and to arbitrary modular relations.

(II) *Proof of knowledge of the discrete logarithm modulo a composite*: In [6, **?**], they apply such PK's to the group of quadratic residues modulo a composite n, $G = QR_n$. Thus the prover needs to convince the verifier that elements he presents are indeed quadratic residues. It is sufficient to execute $PK\{(\alpha) : y^2 = (g^2)^\alpha\}$ instead of $PK\{(\alpha) : y = g^\alpha\}$ [6]. The quantity α is defined as $\log_{g^2} y^2$ which is same as $\log_g y$ in case y is a quadratic residue.

We use the notation $PK^2\{(\alpha) : y = g^\alpha\}$ in the group of quadratic residues modulo a composite, simply.

3 Construction of Pseudonym System

3.1 Procedures

We give an overview of our pseudonym system in this section. The basic system comprises procedures, (1) *System setup*, (2) *Registration of an organization* (*Entry into the system of an organization*), (3) *Registration of a user* ((3-1) *Registration with CA* (*Entry into the system of a user*), (3-2) *Registration with a group*, (3-3) *Registration with an organization*), (4) *Proof the possession of a credential by a user* ((4-1) *Showing a credential with/without identity of an organization*, (4-2) *Transferring a credential with/without identity of an organization*).

In our paper, throughout we assume that users, organizations and group managers are connected by perfect anonymous channels, and each protocol is executed through a secure channel.

1. System setup:
All group managers G_I generate their group secret and public-key pairs $(\mathcal{X}_{G_I}, \mathcal{Y}_{G_I})$ by running GK_{G_I}.

2. Registration with group $\boldsymbol{G}$ of organization O_i:
O_i runs $GK_{(O_i,G)}$, generates a secret and public-key pair $(\mathcal{X}_{(O_i,G)}, \mathcal{Y}_{(O_i,G)})$ by using $\boldsymbol{G}$'s public-key $\mathcal{Y}_G$, and registers $\mathcal{Y}_{(O_i,G)}$. A group manager G publishes a list of public-keys of organizations.

3-1. Registration with CA of user U:
After identification by U, CA checks that U is eligible to join the system. U generates a master secret-key $\mathcal{X}_U$ by running GK_U, both U and CA run $GP\langle U(\mathcal{X}_U), CA\rangle$ to establish U's pseudonym $\mathcal{P}_{(U,CA)}$ which is based on $\mathcal{X}_U$. Then U can receive a credential $\mathcal{C}_{(U,CA)}$, by running $IC\langle U, CA\rangle(\mathcal{P}_{(U,CA)})$.

3-2. Registration with group $\boldsymbol{G}$ of user U:
Both U and G run $GP\langle U(\mathcal{X}_U), G\rangle$ to establish U's pseudonym $\mathcal{P}_{(U,G)}$, and run $TC\langle U, G\rangle$ to demonstrate whether or not U is a valid participant in the system. In TC, U can prove the possession of $\mathcal{C}_{(U,CA)}$ on $\mathcal{P}_{(U,CA)}$ based on $\mathcal{X}_U$ where $\mathcal{P}_{(U,G)} \in GP\langle U(\mathcal{X}_U), G\rangle$. If it is valid, G issues a credential $\mathcal{C}_{(U,G)}$ on $\mathcal{P}_{(U,G)}$ to U by running $IC\langle U, G\rangle$.

3-3. Registration with organization $O_i \in \boldsymbol{G}$ of user U:
Both U and O_i run $GP\langle U(\mathcal{X}_U), O_i\rangle$ to get $\mathcal{P}_{(U,O_i)}$, and run $TC\langle U, O_i\rangle$ to prove the possession of $\mathcal{C}_{(U,G)}$ on $\mathcal{P}_{(U,G)}$ based on $\mathcal{X}_U$. If it is valid, then they run $GV\langle U, O_i\rangle$ to generate a proof of a validity of $\mathcal{P}_{(U,O_i)}$, whose output $\sigma_{(U,O_i)}$ guarantees that U has registered a pseudonym with O_i. Note that $\sigma_{(U,O_i)}$ is the U's private output. After G checks the validity of $\sigma_{(U,O_i)}$, G blindly issues a credential $\mathcal{C}_{(U,O_i)}$ on $\mathcal{P}_{(U,O_i)}$ by running $BIC\langle U(\mathcal{P}_{(U,O_i)}), G\rangle$.

4-1. Showing of a credential on a pseudonym with organization O_i: U chooses a way to show a credential. If U wants to let V know an organization $O_i \in \boldsymbol{G}$ with which U transacts, then both U and V run $SC^+\langle U, V\rangle(\mathcal{Y}_{(O_i,G)})$, which assures that U has $\mathcal{C}_{(U,O_i)}$ on $\mathcal{P}_{(U,O_i)}$ established with O_i. If U does not want to let V know the corresponding organization, both U and V run $SC^-\langle U(\mathcal{Y}_{(O_i,G)}), V\rangle$ which proves the only possession of a credential $\mathcal{C}_{(U,O_i)}$ on $\mathcal{P}_{(U,O_i)}$ without revealing $\mathcal{Y}_{(O_i,G)}, \mathcal{C}_{(U,O_i)}$ and $\mathcal{P}_{(U,O_i)}$.

4-2. Transferring a credential on a pseudonym with organization O_i: Let U register $\mathcal{P}_{(U,O_i)}$ and $\mathcal{P}_{(U,X_j)}$ with an organization $O_i \in \boldsymbol{G}_I$ and $X_j \in \boldsymbol{G}_J$ respectively. Both U and X_j execute $TC^+\langle U, X_j\rangle(\mathcal{Y}_{(O_i,G_I)})$ which assures that U has $\mathcal{C}_{(U,O_i)}$ on $\mathcal{P}_{(U,O_i)}$ based on $\mathcal{X}_U$ where $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$, without revealing $\mathcal{C}_{(U,O_i)}$ and $\mathcal{P}_{(U,O_i)}$. If U does not want to let X_j know the organization O_i, then both U and X_j run $TC^-\langle U(\mathcal{Y}_{(O_i,G_I)}), X_j\rangle$.

3.2 Constructions of pseudonym systems

This section provides constructions of our pseudonym system.

Common system parameter

Security-related system parameters are as follows: the length ℓ_n of the RSA modulus, the integer intervals $\Gamma =]-2^{\ell_\Gamma}, 2^{\ell_\Gamma}[$, $\Delta =]-2^{\ell_\Delta}, 2^{\ell_\Delta}[$, $\Lambda =]2^{\ell_\Lambda}, 2^{\ell_\Lambda+\ell_\Sigma}[$, such that $\ell_\Delta = \epsilon\ell_\Gamma$ and $\ell_\Gamma = 2\ell_n$, where $\epsilon > 1$ is a security parameter, and $2^{\ell_\Lambda} > 2(2^{2\ell_\Gamma} + 2^{\ell_\Gamma} + 2^{\ell_\Delta})$, and $2(2^{\ell_\Sigma}(2^{2\ell_\Gamma} + 2^{\ell_\Delta}) + 2^{\ell_\Delta}) < 2^{\ell_\Lambda}$.

Generation of keys

1. A group manager $G \in \boldsymbol{G}$ chooses random $\ell_n/2$-bit primes p'_G, q'_G such that $p_G := 2p'_G + 1$ and $q_G := 2q'_G + 1$ are prime, sets modulus $n_G := p_G q_G$. It also chooses elements $d_G, e_G, f_G, g_G, h_G \in_R QR_{n_G}$. It stores $\mathcal{X}_G := (p_G, q_G)$ as its secret-keys, and publishes $\mathcal{Y}_G := (n_G, d_G, e_G, f_G, g_G, h_G)$ as its public-key together with a proof that n_G is the product of two safe primes and that the elements d_G, e_G, f_G, g_G and h_G lie indeed in QR_{n_G}.

2. An organization O_i chooses a secret-key $x_{(O_i,G)} \in_R \Gamma$ and sets a corresponding public-key $y_{(O_i,G)} := {g_G}^{x_{(O_i,G)}} \pmod{n_G}$ to register with group $\boldsymbol{G}$. O_i stores $x_{(O_i,G)}$ as a secret-key $\mathcal{X}_{(O_i,G)}$ and publishes $y_{(O_i,G)}$ and its identity $id_{(O_i,G)}$ as O_i's public-keys $\mathcal{Y}_{(O_i,G)}$.

3. A user U chooses a random secret element $x_U \in \Gamma$, and stores it as U's master secret-key $\mathcal{X}_U$ in the system.

Generation of a pseudonym

$GP\langle U, X\rangle$ assures that $\mathcal{P}_{(U,X)} = (N_{(U,X)}, P_{(U,X)})$ is of right form, i.e., $P_{(U,X)} = {g_G}^{x_U} {h_G}^{s_{(U,X)}}$, with $x_U \in \Gamma$ and $s_{(U,X)} \in \Delta$. $N_{(U,X)}$ and $P_{(U,X)}$ are called a nym and its validating tag, respectively. To establish a pseudonym with an entity X, both U and X carry out the following protocol:

1. U chooses $N_1 \in \{0,1\}^k, r_1 \in_R \Delta$ and $r_2, r_3 \in_R \{0,1\}^{2\ell_n}$, sets $c_1 := {d_G}^{r_1} {e_G}^{r_2}$ and $c_2 := {d_G}^{x_U} {e_G}^{r_3}$. U sends N_1, c_1 and c_2 to X, and serves as the prover to verifier X in $PK^2\{(\alpha, \beta, \gamma, \delta) \ : c_1 = {d_G}^{\alpha} {e_G}^{\beta} \wedge c_2 = {d_G}^{\gamma} {e_G}^{\delta}\}$, to prove c_1 and c_2 are generated correctly.

2. X chooses $r \in_R \Delta$, and sends r and N_2 to U.

3. U sets the nym $N_{(U,X)} := N_1 || N_2$, and computes $s_{(U,X)} := (r_1 + r \pmod{2^{\ell_\Delta+1}} + 1)) - 2^{\ell_\Delta} + 1$, and $\tilde{s} = \lfloor (r_1 + r)/(2^{\ell_\Delta+1} - 1) \rfloor$. U sets $P_{(U,X)} := {g_G}^{x_U} {h_G}^{s_{(U,X)}}$ as a validating tag of $N_{(U,X)}$. U sends $P_{(U,X)}$ to X, and shows that it was formed correctly: U sets $c_3 := {d_G}^{\tilde{s}} {e_G}^{r_4}$ for $r_4 \in_R \{0,1\}^{\ell_n}$, sends it to X. Then they engage in

$$
\begin{aligned}
PK^2\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \vartheta, \xi) \ : \ & c_1 = {d_G}^{\alpha} {e_G}^{\beta} \\
& \wedge\ c_2 = {d_G}^{\gamma} {e_G}^{\delta} \\
& \wedge\ c_3 = {d_G}^{\varepsilon} {e_G}^{\zeta} \\
& \wedge\ P_{(U,X)} = {g_G}^{\gamma} {h_G}^{\vartheta} \\
& \wedge\ (c_1 (d_G)^{r - 2^{\ell_\Delta+1}}) / ({c_3}^{2^{\ell_\Delta+1}+1}) = {d_G}^{\vartheta} {e_G}^{\xi} \\
& \wedge\ \gamma \in \Gamma \wedge \vartheta \in \Delta\}.
\end{aligned}
$$

5. X stores $\mathcal{P}_{(U,X)} = (N_{(U,X)}, P_{(U,X)})$ in its database.

6. U stores $(\mathcal{S}_{(U,X)}, \mathcal{P}_{(U,X)}) = (s_{(U,X)}, \{N_{(U,X)}, P_{(U,X)}\})$ in its record with X.

Issue of a credential on a pseudonym with a group

$IC\langle U, G\rangle$ guarantees that a credential on a previously established $\mathcal{P}_{(U,G)}$ is $\mathcal{C}_{(U,G)} = (E_{(U,G)}, C_{(U,G)})$ such that $C_{(U,G)} \equiv (P_{(U,G)} f_G)^{1/E_{(U,G)}} \pmod{n_G}$. To be granted credential, U runs the following protocol with G:

1. U identifies as its owner by $PK^2\{(\alpha, \beta) : P_{(U,G)} = {g_G}^{\alpha} {h_G}^{\beta}\}$ for $\mathcal{P}_{(U,G)}$ in G's database.
2. G chooses a random prime $E_{(U,G)} \in_R \Lambda$, computes $C_{(U,G)} := (P_{(U,G)} f_G)^{1/E_{(U,G)}} \pmod{n_G}$, and sends $E_{(U,G)}$ and $C_{(U,G)}$ to U. Then G stores $\mathcal{C}_{(U,G)} = (E_{(U,G)}, C_{(U,G)})$ as a credential on $\mathcal{P}_{(U,G)}$.
3. U checks if ${C_{(U,G)}}^{E_{(U,G)}} \equiv P_{(U,G)} f_G \pmod{n_G}$ and $E_{(U,G)} \in \Lambda$, and stores $\mathcal{C}_{(U,G)} = (E_{(U,G)}, C_{(U,G)})$ in its record with group $\boldsymbol{G}$.

Showing a credential on a pseudonym with a group

U proves the possession of $\mathcal{C}_{(U,G)} \in IC\langle U, G\rangle$ by running SC. Both U and V engage in the following protocol:

1. U sets $c_1 := C_{(U,G)} {e_G}^{r_1}$ and $c_2 := {e_G}^{r_1} {d_G}^{r_2}$ for $r_1, r_2 \in_R \{0,1\}^{2\ell_n}$, and sends c_1 and c_2 to V,
2. U engages with V in

$$\begin{aligned} PK^2\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \xi) \,:\, & f_G = {c_1}^{\alpha} / {g_G}^{\beta} {h_G}^{\gamma} {e_G}^{\delta} \\ & \wedge\, c_2 = {e_G}^{\varepsilon} {d_G}^{\zeta} \\ & \wedge\, 1 = {c_2}^{\alpha} / {e_G}^{\delta} {d_G}^{\xi} \\ & \wedge\, \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}. \end{aligned}$$

Transferring a credential on a pseudonym with a group

TC assures that U owns $\mathcal{C}_{(U,G_I)}$ on $\mathcal{P}_{(U,G_I)}$ based on $\mathcal{X}_U$ where $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$. U proves it by running TC with $X_j \in \boldsymbol{G}_J$ with whom U has established $\mathcal{P}_{(U,X_j)}$:

1. U sets $c_1 := C_{(U,G_I)} {e_G}^{r_1}$ and $c_2 := {e_{G_I}}^{r_1} {d_{G_I}}^{r_2}$ for $r_1, r_2 \in_R \{0,1\}^{2\ell_n}$, and sends c_1 and c_2 to X_j,
2. U engages with X_i in

$$\begin{aligned} PK^2\{(\alpha, \beta, \gamma, \delta, \varepsilon, \zeta, \xi, \eta) \,:\, & f_{G_I} = {c_1}^{\alpha} / {g_{G_I}}^{\beta} {h_{G_I}}^{\gamma} {e_{G_I}}^{\delta} \\ & \wedge\, c_2 = {e_{G_I}}^{\varepsilon} {d_{G_I}}^{\zeta} \\ & \wedge\, 1 = {c_2}^{\alpha} / {e_{G_I}}^{\delta} {d_{G_I}}^{\xi} \\ & \wedge\, P_{(U,X_j)} = {g_{G_J}}^{\beta} {h_{G_J}}^{\eta} \\ & \wedge\, \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}, \end{aligned}$$

for $\mathcal{P}_{(U,X_j)}$ in X_j's database.

Generation of a proof of pseudonym's validity

GV guarantees that U's output $\sigma_{(U,O_i)}$ is independent of O_i's view of the conversation. In order to generate a signature on $\mathcal{P}_{(U,O_i)}$, both U and O_i run GV:

1. U identifies as its owner by $PK^2\{(\alpha,\beta) : P_{(U,O_i)} = {g_G}^{\alpha}{h_G}^{\beta}\}$, for $\mathcal{P}_{(U,O_i)}$ in O_i's database.
2. O_i generates $Q_{(U,O_i)} := {P_{(U,O_i)}}^{x_{(O_i,G)}}$, $t_1 := {g_G}^r$ and $t_2 := {P_{(U,O_i)}}^r$ for $r \in_R \{0,1\}^{2\ell_n}$, and sends $Q_{(U,O_i)}, t_1$ and t_2 to U.
3. U chooses r_1, r_2 and $r_3 \in_R \{0,1\}^{2\ell_n}$ and computes ${t_1}' := t_1 {g_G}^{r_1} {y_{(O_i,G)}}^{r_2}$, ${t_2}' := (t_2 {P_{(U,O_i)}}^{r_1} {Q_{(U,O_i)}}^{r_2})^{r_3}$, ${P'}_{(U,O_i)} := {P_{(U,O_i)}}^{r_3}$ and ${Q'}_{(U,O_i)} := {Q_{(U,O_i)}}^{r_3}$. Then U sets $e' := \mathcal{H}(g_G, y_{(O_i,G)}, {P'}_{(U,O_i)}, {Q'}_{(U,O_i)}, {t_1}', {t_2}')$, sends $e := e' - r_2$ to O_i.
4. O_i computes $s := r - e x_{(O_i,G)}$ and sends it to U.
5. U checks if $t_1 = {g_G}^s {y_{(O_i,G)}}^e$, $t_2 = {P_{(U,O_i)}}^s {Q_{(U,O_i)}}^e$, and sets $s' := s + r_1$. Then U stores $\sigma_{(U,O_i)} := (e', s', {P'}_{(U,O_i)}, {Q'}_{(U,O_i)})$ as a proof of a validity of $\mathcal{P}_{(U,O_i)}$, and keeps r_3 secretly until U gets a credential on $\mathcal{P}_{(U,O_i)}$.

Issue of a credential on a pseudonym with an organization

$BIC\langle U, G\rangle$ guarantees that a credential on $\mathcal{P}_{(U,O_i)}$ is $\mathcal{C}_{(U,O_i)} = (E_{(U,O_i)}, C_{(U,O_i)})$ such that $C_{(U,O_i)} \equiv (P_{(U,O_i)} {d_G}^{id_{(O_i,G)}} f_G)^{1/E_{(U,O_i)}}$. BIC establishes $\mathcal{C}_{(U,O_i)}$ without revealing anything more than the fact that U has registered with O_i to G. Such a credential can be granted by using the blind RSA-signature [1] in the following protocol:

1. U chooses a prime $E_{(U,O_i)} \in_R \Lambda$ and $r \in_R \mathbb{Z}_{n_G}$, and generates $c := r^{E_{(U,O_i)}} P_{(U,O_i)} {d_G}^{id_{(O_i,G)}} f_G$. Then U sends $c, E_{(U,O_i)}$ and $\sigma_{(U,O_i)}$. Furthermore U must show that $\sigma_{(U,O_i)}$ was generated to U and c was generated correctly: U computes $c_1 := r {e_G}^{r_1}$ for $r_1 \in_R \{0,1\}^{2\ell_n}$, and engages with G in

$$\begin{aligned} PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta) \,:\, & P_{(U,G)} = {g_G}^{\alpha}{h_G}^{\beta} \\ & \wedge\ 1 = {P_{(U,G)}}^{\gamma} / {g_G}^{\delta}{h_G}^{\varepsilon} \\ & \wedge\ {P'}_{(U,O_i)} = {g_G}^{\delta}{h_G}^{\zeta} \\ & \wedge\ {P'}_{(U,O_i)} = c^{\gamma}({e_G}^{E_{(U,O_i)}})^{\xi} / ({c_1}^{E_{(U,O_i)}} {d_G}^{id_{(U,O_i)}} f_G)^{\gamma} \\ & \wedge\ \alpha \in \Gamma, \beta \in \Delta\}, \end{aligned}$$

for $\mathcal{P}_{(U,G)}$ in G's database.
2. O_i checks if σ is valid: if $e' = \mathcal{H}(g_G, y_{(O_i,G)}, {P'}_{(U,O_i)}, {Q'}_{(U,O_i)}, \tilde{t_1}, \tilde{t_2})$ where $\tilde{t_1} = {g_G}^{s'} {y_{(O_i,G)}}^{e'}$, $\tilde{t_2} = {{P'}_{(U,O_i)}}^{s'} {{Q'}_{(U,O_i)}}^{e'}$, and $y_{(O_i,G)}$ is in $\boldsymbol{G}$'s public-key list. Then O_i computes $c' := c^{1/E_{(U,O_i)}}$ and sends it to U.
3. U sets $C_{(U,O_i)} := c'/r$. Then U checks if ${C_{(U,O_i)}}^{E_{(U,O_i)}} \equiv P_{(U,O_i)} {d_G}^{id_{(O_i,G)}} f_G \pmod{n_G}$, and stores $(E_{(U,O_i)}, C_{(U,O_i)})$ in its record with organization O_i.

Showing a credential with identity of an organization

To prove the possession of $\mathcal{C}_{(U,O_i)} \in BIC\langle U, G\rangle$, both U and V run SC^+. They engage in the following protocol:

1. U sets $c_1 := C_{(U,O_i)} {e_G}^{r_1}$ and $c_2 := {e_G}^{r_1} {d_G}^{r_2}$ for $r_1, r_2 \in_R \{0,1\}^{2\ell_n}$, and sends c_1 and c_2 to V,
2. U engages with V in

$$\begin{aligned} PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi) \,:\, & f_G {d_G}^{id_{(O_i,G)}} = {c_1}^{\alpha} / {g_G}^{\beta} {h_G}^{\gamma} {e_G}^{\delta} \\ & \wedge c_2 = {e_G}^{\varepsilon} {d_G}^{\zeta} \\ & \wedge 1 = {c_2}^{\alpha} / {e_G}^{\delta} {d_G}^{\xi} \\ & \wedge \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}. \end{aligned}$$

Showing a credential without identity of an organization

In order to prove the possession of a credential generated by running $BIC\langle U, G\rangle$, both U and V run SC^-. They engage in the following protocol:

1. U sets $c_1 := C_{(U,O_i)} {e_G}^{r_1}$ and $c_2 := {e_G}^{r_1} {d_G}^{r_2}$ for $r_1, r_2 \in_R \{0,1\}^{2\ell_n}$, and sends c_1 and c_2 to V,
2. U engages with V in

$$\begin{aligned} PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta) \,:\, & f_G = {c_1}^{\alpha} / {g_G}^{\beta} {h_G}^{\gamma} {d_G}^{\delta} {e_G}^{\varepsilon} \\ & \wedge c_2 = {e_G}^{\zeta} {d_G}^{\xi} \\ & \wedge 1 = {c_2}^{\alpha} / {e_G}^{\varepsilon} {d_G}^{\eta} \\ & \wedge \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}. \end{aligned}$$

Transferring a credential with identity of an organization

In TC^+, U proves the possession of $\mathcal{C}_{(U,O_i)}$ on $\mathcal{P}_{(U,O_i)}$ based on $\mathcal{X}_U$ where $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$ to $X_j \in \boldsymbol{G}_J$:

1. U sets $c_1 := C_{(U,O_i)} {e_{G_I}}^{r_1}$ and $c_2 := {e_{G_I}}^{r_1} {d_{G_I}}^{r_2}$ for $r_1, r_2 \in_R \{0,1\}^{2\ell_n}$, and sends c_1 and c_2 to X_j,
2. U engages with X_j in

$$\begin{aligned} PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta) \,:\, & f_{G_I} {d_{G_I}}^{id_{(O_i,G_I)}} = {c_1}^{\alpha} / {g_{G_I}}^{\beta} {h_{G_I}}^{\gamma} {e_{G_I}}^{\delta} \\ & \wedge c_2 = {e_{G_I}}^{\varepsilon} {d_{G_I}}^{\zeta} \\ & \wedge 1 = {c_2}^{\alpha} / {e_{G_I}}^{\delta} {d_{G_I}}^{\xi} \\ & \wedge P_{(U,X_j)} = {g_{G_J}}^{\beta} {h_{G_J}}^{\eta} \\ & \wedge \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}, \end{aligned}$$

for $\mathcal{P}_{(U,X_j)}$ in X_j's database.

Transferring a credential without identity of an organization

U proves the possession of a credential generated by running $BIC\langle U, G_I\rangle$ on a pseudonym based on $\mathcal{X}_U$ where $\mathcal{P}_{(U,X_j)} \in GP\langle U(\mathcal{X}_U), X_j\rangle$:

1. U sets $c_1 := C_{(U,O_i)} {e_{G_I}}^{r_1}$ and $c_2 := {e_{G_I}}^{r_1} {d_{G_I}}^{r_2}$ for $r_1, r_2 \in_R \{0,1\}^{2\ell_n}$, and sends c_1 and c_2 to X_j,

2. U engages with X_j in

$$\begin{aligned} PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta,\varphi) \,:\, & f_{G_I} = {c_1}^{\alpha}/{g_{G_I}}^{\beta}{h_{G_I}}^{\gamma}{d_{G_I}}^{\delta}{e_{G_I}}^{\varepsilon} \\ & \wedge\, c_2 = {e_{G_I}}^{\zeta}{d_{G_I}}^{\xi} \\ & \wedge\, 1 = {c_2}^{\alpha}/{e_{G_I}}^{\varepsilon}{d_{G_I}}^{\eta} \\ & \wedge\, P_{(U,X_j)} = {g_{G_J}}^{\beta}{h_{G_J}}^{\varphi} \\ & \wedge\, \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}, \end{aligned}$$

for $P_{(U,X_j)}$ in X_j's database.

4 Proof of Security for Our Scheme

In this section, we assess the security of our pseudonym system. Under the strong RSA assumption and the decisional Diffie-Hellman assumption modulo a safe prime product, the following technical lemmas about the protocols described are stated here:

Lemma 3 *The* $PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\vartheta,\xi) : c_1 = {d_G}^{\alpha}{e_G}^{\beta} \wedge c_2 = {d_G}^{\gamma}{e_G}^{\delta} \wedge c_3 = {d_G}^{\varepsilon}{e_G}^{\zeta} \wedge P_{(U,X)} = {g_G}^{\gamma}{h_G}^{\vartheta} \wedge (c_1(d_G)^{r-2^{\ell_\Delta+1}})/({c_3}^{2^{\ell_\Delta+1}+1}) = {d_G}^{\vartheta}{e_G}^{\xi} \wedge \gamma \in \Gamma \wedge \vartheta \in \Delta\}$ *in GP is a statistical zero-knowledge proof of knowledge of the correctly formed values* $x_U, s_{(U,X)}$ *that correspond to a pseudonym validating tag* $P_{(U,X)}$.

Lemma 4 *The* PK^2 *protocols in* SC, TC, SC^+, SC^-, TC^+ *and* TC^- *are a statistical zero-knowledge proof of knowledge of the prover's master secret-key, corresponding secret information(s) and credential in right form.*

The proofs of these Lemmas is closely related to Lemma 1, 2 and 3 of [6], we only prove the security of the PK^2 protocol in TC^- here. The other proofs can easily inferred from the following.

Lemma 5 *The* $PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta,\varphi) : f_{G_I} = {c_1}^{\alpha}/{g_{G_I}}^{\beta}{h_{G_I}}^{\gamma}{d_{G_I}}^{\delta}{e_{G_I}}^{\varepsilon} \wedge c_2 = {e_{G_I}}^{\zeta}{d_{G_I}}^{\xi} \wedge 1 = {c_2}^{\alpha}/{e_{G_I}}^{\varepsilon}{d_{G_I}}^{\eta} \wedge P_{(U,X_j)} = {g_{G_J}}^{\beta}{h_{G_J}}^{\varphi} \wedge \alpha \in \Lambda \wedge \beta \in \Gamma \wedge \gamma \in \Delta\}$ *in* TC^- *is a statistical zero-knowledge proof of knowledge of the values* $x \in \Gamma, s_1, s_2 \in \Delta, E \in \Lambda, C$, *and* y *such that* $P_{(U,X_j)} = {g_{G_J}}^{x}{h_{G_J}}^{s_1} \pmod{n_{G_J}}$, *and* $C^E = {g_{G_I}}^{x}{h_{G_I}}^{s_2}{d_{G_I}}^{y} f_{G_I} \pmod{n_{G_I}}$.

Proof. By the properties of the PK^2 and the RSA assumption, the knowledge extractor can produce values $\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta,\varphi$ such that the statement after the colon holds. As $c_2 = {e_{G_I}}^{\zeta}{d_{G_I}}^{\xi}$ and $1 = {c_2}^{\alpha}/{e_{G_I}}^{\varepsilon}{d_{G_I}}^{\eta}$ from which we conclude that $\zeta\alpha = \varepsilon \pmod{ord(e_{G_I})}$, we have ${c_1}^{\alpha}/{e_{G_I}}^{\varepsilon} = {g_{G_I}}^{\beta}{h_{G_I}}^{\gamma}{d_{G_I}}^{\delta} f_G = (c_1/{e_{G_I}}^{\zeta})^{\alpha}$, where $\alpha \in \Lambda, \beta \in \Gamma$ and $\gamma \in \Delta$. It follows that U must know a valid credential $c_1/{e_{G_I}}^{\zeta}$ on a pseudonym. Furthermore, from $P_{(U,X_j)} = {g_{G_J}}^{\beta}{h_{G_J}}^{\eta}$, it guarantees that both pseudonym $P_{(U,X_j)}$ and a pseudonym registered with O_i are based on the same master secret key.

Lemma 6 *The* $PK^2\{(\alpha,\beta,\gamma,\delta,\varepsilon,\zeta,\xi,\eta) : P_{(U,G)} = {g_G}^{\alpha}{h_G}^{\beta} \wedge 1 = {P_{(U,G)}}^{\gamma}/{g_G}^{\delta}{h_G}^{\varepsilon} \wedge P'_{(U,O_i)} = {g_G}^{\delta}{h_G}^{\zeta} \wedge P'_{(U,O_i)} = c^{\gamma}({e_G}^{E_{(U,O_i)}})^{\xi}/({c_1}^{E_{(U,O_i)}}{d_G}^{id_{(U,O_i)}}f_G)^{\gamma} \wedge \alpha \in \Gamma, \beta \in \Delta\}$ *in BIC is a statistical zero-knowledge proof of knowledge of the values* $x \in \Gamma, s \in \Delta, r$, *and* r_3 *such that* $P_{(U,G)} = {g_G}^{x}{h_G}^{s}$, $c = r^{E_{(U,O_i)}}P_{(U,O_i)}{d_G}^{id_{(O_i,G)}}f_G$ *and* $P'_{(U,O_i)} = {P_{(U,O_i)}}^{r_3}$.

Proof. In the statement after the colon, $P_{(U,G)} = {g_G}^{\alpha}{h_G}^{\beta}$ and $1 = {P_{(U,G)}}^{\gamma}/{g_G}^{\delta}{h_G}^{\varepsilon}$ from which we conclude $\alpha\gamma \equiv \delta \pmod{ord(g_G)}$. From $P'_{(U,O_i)} = c^{\gamma}({e_G}^{E_{(U,O_i)}})^{\xi}/({c_1}^{E_{(U,O_i)}}{d_G}^{id_{(U,O_i)}}f_G)^{\gamma}$, we have $c^{\gamma} = {g_G}^{\delta}{h_G}^{\zeta}({c_1}^{E_{(U,O_i)}}{d_G}^{id_{(O_i,G)}}f_G)^{\gamma}/({e_G}^{E_{(U,O_i)}})^{\xi} = \{(c_1/{e_G}^{\xi/\gamma})^{E_{(U,O_i)}}{g_G}^{\alpha}{h_G}^{\zeta/\gamma}{d_G}^{id_{(O_i,G)}}f_G\}^{\gamma}$. As $\alpha \in \Gamma$ and $\beta \in \Delta$, c is formed collectly, by using the same key underlying $P_{(U,G)}$.

4.1 Description of the simulator

We have to describe simulator $\mathcal{S}$ for our scheme and then show that it satisfies *Definition 1*. The parties the adversary $\mathcal{A}$ controls are subsumed into a single party. We only describe the simulator for the adversary.

Setup.

For the group manager $G \in \boldsymbol{G}$ and the organization $O \in \boldsymbol{G}$ not controlled by $\mathcal{A}$, $\mathcal{S}$ sets up their secret and public-key $(\mathcal{X}_G, \mathcal{Y}_G)$, and $(\mathcal{X}_{(O_i,G)}, \mathcal{Y}_{(O_i,G)})$ as dictated by the protocol. Furthermore, $\mathcal{S}$ creates an $archive_G$ or $archive_O$ where it will record the credentials of users controlled by $\mathcal{A}$ with the group manager or the organization. It also initialized a list of users controlled by $\mathcal{A}$, $list_{\mathcal{A}}$.

Generation of a pseudonym with a group.

A user establishes a pseudonym with a group manager G:

(I) If a user is controlled by $\mathcal{A}$, (i) $\mathcal{S}$ uses the knowledge extractor of Lemma 3 to discover the user's master secret key x and the secret values s, (i-1) If $x \notin list_{\mathcal{A}}$, $\mathcal{S}$ creates a new user U with login name L_U, and obtains a pseudonym $N_{(U,G)}$, and a key K_U corresponding to L_U by interaction with T. Denote (x, s) by $(x_U, s_{(U,G)})$, $\mathcal{S}$ stores $(U, L_U, x_U, K_U, N_{(U,G)}, s_{(U,G)})$ in $list_{\mathcal{A}}$, (ii-2) If $x \in list_{\mathcal{A}}$, $\mathcal{S}$ obtains $N_{(U,G)}$ for this user U corresponding to x by interaction with T, and adds $N_{(U,G)}, s_{(U,G)} := s$ to U's record.

(II) If a group manager G is controlled by $\mathcal{A}$, $\mathcal{S}$ will use the zero-knowledge simulator from Lemma 3 to furnish the $\mathcal{A}$'s view of the protocol.

Issue a credential on a pseudonym with a group.

A user requests a group manager G to issue a credential:

(I) If a user is controlled by $\mathcal{A}$, (i) upon receiving a message from T, $\mathcal{S}$ runs the knowledge extractor for the proof of knowledge of step 1 of IC, to determine the value x and s. For N corresponding to (x, s), (i-1) if $N \notin list_{\mathcal{A}}$, then $\mathcal{S}$ refuses to grant a credential. (i-2) If $N \in list_{\mathcal{A}}$, then $\mathcal{S}$ issued the correct E and C by interaction with T. $\mathcal{S}$ stores the values $(x_U, s_{(U,G)}, E_{(U,G)}, C_{(U,G)}) := (x, s, E, C)$ in $archive_G$.

(II) If a group manager G is controlled by $\mathcal{A}$, $\mathcal{S}$ will run the zero-knowledge simulator for step 1 of IC, and continue the protocol as U would. If the user accepts, then $\mathcal{S}$ informs T that the credential was granted.

Generation of a pseudonym with an organization.

A user establishes a pseudonym with an organization $O \in \boldsymbol{G}$:

This part of the simulator can easily be inferred from the part for the above *Generation of a pseudonym with a group.*

Generation of a proof of pseudonym's validity.

A user requests an organization O to grant a proof of pseudonym's validity: (I) If a user is controlled by $\mathcal{A}$, (i) $\mathcal{S}$ uses the knowledge extractor for PK of step 1 of GV to discover the user's key x and the value s. For N corresponding to (x, s), (i-1) If $N \notin list_{\mathcal{A}}$, $\mathcal{S}$ refuses to grant a proof of pseudonym's validity. (i-2) If $N \in list_{\mathcal{A}}$, $\mathcal{S}$ grants σ by interaction with T.

(II) If an organization O is controlled by $\mathcal{A}$, $\mathcal{S}$ will run the zero-knowledge simulator for step 1 of GV, and continue the protocol as U would.

Issue a credential on a pseudonym with an organization.

A user requests a group manager $G \in \boldsymbol{G}$ to issue a credential with an organization $O \in \boldsymbol{G}$:

(I) If a user is controlled by $\mathcal{A}$, (i) upon receiving a message from T, $\mathcal{S}$ runs the knowledge extractor for the proof of knowledge of step 1 of BIC to extract the value $x, s_{(U,G)}, s_{(U,O)}$ and r. (i-1) If $(x, s_{(U,G)}) \notin archive_G$, then $\mathcal{S}$ refuses to grant a credential, (i-2) If $(x, s_{(U,G)}) \in archive_G$, then $\mathcal{S}$ issues the correct c' corresponding to E by executing the rest of the G's side of it. $\mathcal{S}$ determines C by c'/r. It stores the values $(x, s_{(U,O)}, C, E)$ in $archive_O$.

(II) If a user is controlled by $\mathcal{A}$ and an organization $O \in \boldsymbol{G}$ is dishonest, (i) upon receiving a message from T, $\mathcal{S}$ runs the knowledge extractor for the proof of knowledge of step 1 of BIC, to extract the value $x, s_{(U,G)}, s$ and r. $\mathcal{S}$ looks at $archive_O$: (i-1) If $(x, s) \in archive_O$, $\mathcal{S}$ denotes this user by U, (i-2) If $(x, s) \notin archive_O$, let U be the user with x. $\mathcal{S}$ obtains $N_{(U,O)}$ by interaction with T, (ii) $\mathcal{S}$ looks at $archive_G$: (ii-1) If $(x, s_{(U,G)}) \notin archive_G$, then $\mathcal{S}$ refuses to grant a credential, (ii-2) If $(x, s_{(U,G)}) \in archive_G$, then $\mathcal{S}$ issues the correct c' corresponding to E by executing the rest of the G's side of it. $\mathcal{S}$ determines C by c'/r. It stores the values $(x, s_{(U,O)}, C, E)$ in $archive_O$.

(III) If an issuing group manager $G \in \boldsymbol{G}$ controlled by $\mathcal{A}$, $\mathcal{S}$ will run the zero-knowledge simulator for step 1 of BIC, and execute the rest of the user's side of it. If the user accepts, then $\mathcal{S}$ informs T that the credential was granted.

Showing a credential with identity of an organization

Showing a credential without identity of an organization

Transferring a credential with identity of an organization

These parts of the simulator can easily be inferred from the part for *Transferring a credential without identity of an organization* that follows.

Transferring a credential without identity of an organization.

A user wants to show ownership of a credential of a pseudonym with some organization in a group $\boldsymbol{G}_I$ to an organization $O_j \in \boldsymbol{G}_J$:

(I) If a user is controlled by $\mathcal{A}$, (i) $\mathcal{S}$ runs O_j's part of TC^-, and extracts the values $x, s_{(U,O_i)}, s_{(U,O_j)}, E, C$ and $y_{(O_i,G_I)}$ with the knowledge extractor of Lemma 5. (i-1) if $(x, s_{(U,O_i)}, E, C) \notin archive_{O_i}$, $\mathcal{S}$ rejects, (i-2) If

$(x, s_{(U,O_i)}, E, C) \in archive_{O_i}$, $\mathcal{S}$ communicates with T for transferring a credential by U.
(II) If a user is controlled by $\mathcal{A}$ and an issuing group manager G_I is dishonest, (i) $\mathcal{S}$ runs O_j' side of CT^- with the knowledge extractor of Lemma 5 to obtain the values $x, s, s_{(U,O_j)}, E, C$ and y, let O_i be an organization whose public-key is y. (i-1) If O_j's side of the protocol reject, it does nothing, (i-2) Otherwise: (2-A-a) If $x \in archive_{O_i}$, denote this user by U, (2-A-b) If $x \notin archive_{O_i}$, let U be the user with x, and $\mathcal{S}$ obtain $N_{(U,O_i)}$ by interaction with T. (2-B) If $(E, C) \notin archive_{O_i}$, then $\mathcal{S}$ runs BIC, adds the output to U's record. (2-C) $\mathcal{S}$ communicates with T for transferring a credential by U.
(III) If a verification organization O_j is controlled by $\mathcal{A}$, $\mathcal{S}$ runs the zero-knowledge simulator of Lemma 5 to do that.

4.2 Proof of Successful Simulation

We show that our simulator fails with negligible probability only. We show in the following lemma that a tuple (x, s, E, C) the knowledge of which is essential for proving possession of a credential, is unforgeable even under an adaptive attack. As these proofs can be found in [6], we leave out the proofs.

Lemma 7 *Under the strong RSA assumption and the discrete logarithm assumption modulo a safe prime product, if a polynomially bounded adversary succeeds in proving ownership of a valid credential record (P, E, C) with a group $\boldsymbol{G}$, then this credential record was created by running GP, IC and TC with a group manager $G \in \boldsymbol{G}$.*

Lemma 8 *Under the strong RSA assumption, the discrete logarithm assumption modulo a safe prime product and , if a polynomially bounded adversary succeeds in proving ownership of a valid credential record (P, E, C) with an organization $O \in \boldsymbol{G}$, then this credential record was created by running GP and TC with an organization $O \in \boldsymbol{G}$, BIC with a group manager $G \in \boldsymbol{G}$.*

The statistical zero-knowledge property of the underlying protocols gives us Lemma 9 which in turn implies Theorem 10.

Lemma 9 *The view of the adversary in the real protocol is statistically close to his view in the simulation.*

Theorem 10 *Under the strong RSA assumption, the decisional Diffie-Hellman assumption modulo a safe prime product, and the assumption that factoring is hard, our pseudonym system described above is secure.*

5 Conclusion

This paper presents an anonymity-enhanced pseudonym system; a user can select a way to prove the possession of a credential on a pseudonym with an organization. We can add a mechanism: *global anonymity revocation* for discovering the

identity of a user whose transactions are illegal, or *local anonymity revocation* for revealing a pseudonym of a user who misuses the credential, in the same way as [6].

References

1. D.Chaum, "Security without identification: Transaction systems to make big brother obsolete", Communications of the ACM, vol. 28, 1030–1044, 1985
2. D.Chaum and J.-H.Evertse, "A secure and privacy - protecting protocol for transmitting personal information between organizations", Proceedings of CRYPTO'86, vol. 263, 118–167, Springer Verlag, 1987
3. L.Chen, "Access with pseudonyms", Cryptography: Policy and Algorithms, vol. 1029, 232–243, Springer Verlag, 1995
4. I.B.Damgard, "Payment systems and credential mechanism with provable security against abuse by individuals", Proceedings of CRYPTO'88, vol. 403, 328–335, Springer Verlag, 1990
5. A.Lysyanskaya and R.Rivest and A.Sahai and S.Wolf, "Pseudonym Systems", Selected Areas in Cryptography, vol. 1758, Springer Verlag, 1999
6. J.Camenisch and A.Lysyanskaya, "Efficient non-transferable anonymous multishow credential system with optional anonymity revocation", Proceedings of EUROCRYPT 2001, vol. 2045, 93–118, Springer Verlag, 2001
7. J.Camenisch and A.Lysyanskaya, "Dynamic accumulators and application to efficient revocation of anonymous credentials", Proceedings of CRYPTO 2002, vol. 2442, 61–76, Springer Verlag, 2002
8. J.Camenisch and E.V.Herreweghen, "Design and implementation of the idemix anonymous credential system", ACM CCS'02, 2002
9. G.Ateniese and J.Camenisch and M.Joye and G.Tsudik, "A practical and provably secure coalition-resistant group signature scheme", Proceedings of CRYPTO 2000, vol. 1880, 255–270, Springer Verlag, 2000
10. C.P.Schnorr, "Efficient signature generation for smart cards", Journal of Cryptology, vol. 4, 239–252, 1991
11. A.Fiat and A.Shamir, "How to prove yourself: Practical solution to identification and signature problems", Proceedings of CRYPTO'86, vol. 263, 186–194, Springer Verlag, 1987
12. E.Fujisaki and T.Okamoto, "Statistical zero knowledge protocols to prove modular polynomial relations", Proceedings of CRYPTO'97, vol. 1294, 16–30, Springer Verlag, 1997
13. J.Camenisch and M.Stadler, "Efficient group signature schemes for large groups", Proceedings of CRYPTO'97, vol. 1294, 410–424, Springer Verlag, 1997
14. R.Cramer and V.Shoup, "Signature schemes based on the strong RSA assumption", Proceedings of 6th ACM Conference on Computer and Communications Security, 46–52, ACM press, 1999
15. M.Bellare and C.Namprempre and D.Pointcheval and M.Semanko, "The Power of RSA Inversion Oracles and the Security of Chaum's RSA-Based Blind Signature Scheme", Proceedings of Financial Cryptography 2001, vol. 2339, 319–338, Springer Verlag, 2001