PPDCP-ABE: Privacy-Preserving Decentralized Ciphertext-Policy Attribute-Based Encryption

Jinguang Han[1], Willy Susilo[2], Yi Mu[2], Jianying Zhou[3], Man Ho Au[2]

[1] Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing, Jiangsu 210003, China
[2] School of Computer Science and Software Engineering, University of Wollongong, Wollongong, NSW 2522, Australia
[3] Infocomm Security Department, Institute for Infocomm Research, 1 Fusionopolis Way, Singapore 138632, Singapore
jghan22@gmail.com, {wsusilo,ymu,aau}@uow.edu.au, jyzhou@i2r.a-star.edu.sg

Abstract. Cipher-policy attribute-based encryption (CP-ABE) is a more efficient and flexible encryption system as the encryptor can control the access structure when encrypting a message. In this paper, we propose a privacy-preserving decentralized CP-ABE (PPDCP-ABE) scheme where the central authority is not required, namely each authority can work independently without the cooperation to initialize the system. Meanwhile, a user can obtain secret keys from multiple authorities without releasing his global identifier (GID) and attributes to them. This is contrasted to the previous privacy-preserving multi-authority ABE (PPMA-ABE) schemes where a user can obtain secret keys from multiple authorities with them knowing his attributes and a central authority is required. However, some sensitive attributes can also release the user's identity information. Hence, contemporary PPMA-ABE schemes cannot fully protect users' privacy as multiple authorities can cooperate to identifier a user by collecting and analyzing his attributes. Therefore, it remains a challenging and important work to construct a PPMA-ABE scheme where the central authority is not required and both the identifiers and the attributes are considered.

Keywords: CP-ABE, decentralization, privacy.

1 Introduction

In network society, users can be identified by their distinct attributes. For example, European electronic identity cards often contain the attributes: nationality, sex, civil status, hair and eye color, and applicable minority status. These attributes are either binary or discrete numbers from a pre-defined finite sets [1]. Especially, they are very privacy-sensitive and require a selective disclosure of one while hiding others completely; otherwise, a user can be identified and impersonated by collecting and analyzing his attributes.

In practical applications, we often share data with some expressive attributes without knowing who will receive it. To resolve this problem, Sahai and Waters

M. Kutyłowski and J. Vaidya (Eds.): ESORICS 2014, Part II, LNCS 8713, pp. 73–90, 2014.

[2] introduced the seminal concept of attribute-based encryption (ABE). In this new encryption system, there is a central authority who monitors the universal attributes and distributes secrete keys to users accordingly. A user can decrypt a ciphertext if and only if there is a match between the attributes which he holds and the attributes listed in the ciphertext. Since it can protect the confidentiality of sensitive data and express flexible access control, ABE schemes have been focused extensively [3–8].

To reduce the trust on the central authority, Chase [9] proposed a multi-authority ABE (MA-ABE) scheme where multiple authorities must cooperate with the central authority to initialize the system. Then, Lewko and Waters [10] proposed a new MA-ABE scheme called decentralized CP-ABE (DCP-ABE) where multiple authorities can work independently without a central authority or any cooperation among them.

1.1 Privacy in Multi-Authority Attribute-Based Encryption

In an MA-ABE scheme, malicious users may combine their secret keys to create a new secret key if the multiple authorities work independently [9]. For example, suppose that there is a ciphertext which can be decrypted by the attributes monitored by the authorities A_1 and A_2. If Alice obtains secret keys from A_1 and Bob obtains secret keys from A_2, they can collaborate to decrypt the ciphertext. To overcome this hurdle, each user in the system [9] must be designated an unique global identifier (GID) which is known by each authority. When generating secret keys for the user, the authorities tie them to his GID.

Privacy issues in MA-ABE are the primary concern of users as the authorities can personate the target user if they know his attributes. Some schemes towards solving this problem have been proposed, but they cannot provide a complete solution, because, in all these schemes, only the privacy of the GID has been considered. Currently, there is no any scheme addressing the privacy issue of the attributes in MA-ABE schemes. However, it is extremely important as a user can be identified by some sensitive attributes. For example, suppose that the Head of the Department of Computer Science is Bob. Given two sets of attributes S_1={Position="Header", Department="CS", Sex="Male"} and S_2= {Position="PhD Student", Department="CS", Sex="Male"}, we can guess S_1 is the attributes of Bob even if we do not know his GID. This clearly shows that controlled release of sensitive attributes is necessary.

1.2 Our Contributions

In this paper, we propose a privacy-preserving decentralized CP-ABE (PPDCP-ABE) scheme. In our scheme, any authority can dynamically join or leave the system, and there is no any requirement for the central authority or interactions among multiple authorities. As a notable feature, each authority can work independently, while other authorities do not need to change their secret keys and reinitialize the system when an authority joins or leaves the system. Each

authority monitors a set of attributes and distributes secret keys to users accordingly. To resist the collusion attacks, user's secret keys are tied to his GID. Especially, a user can obtain secret keys for his attributes from multiple authorities without revealing any information about his GID and attributes to the authorities. Therefore, it provides stronger privacy compared to the previous PPMA-ABE schemes where only the identifier is protected. To encrypt a message, the encryptor selects an access structure for each authority and encrypts the message under them so that only the users whose attributes satisfy all the access structures can decrypt the ciphertext and obtain the plaintext. Compared to the existing decentralized ABE scheme [10] which was constructed in the random oracle model, our scheme is designed in the standard model. To the best of our knowledge, it is the *first* PPDCP-ABE scheme where both the identifiers and attributes are considered.

1.3 Organization

The remainder of this paper is organized as follows. In Section 2, the related work is introduced. We describe the preliminaries which are used throughout this paper in Section 3. In Section 4, we first construct a DCP-ABE scheme, and then propose a privacy-preserving key extract algorithm for it. Finally, Section 5 concludes this paper.

2 Related Work

In this section, the related work is introduced.

2.1 Attribute-Based Encryption

Introduced by Sahai and Waters [2], attribute-based encryption (ABE) is a new encryption system where both the ciphertext and the secret key are labeled with a set of attributes. A user can decrypt a ciphertext if and only if there is a match between the attributes listed in the ciphertext and the attributes held by the user. Currently, ABE schemes can be classified into two types: key-policy ABE (KP-ABE) and cipher-policy ABE (CP-ABE).

KP-ABE. In these schemes, an access structure is embedded in the secret keys, while the ciphertext is associated with a set of attributes [2, 9, 11, 5, 6, 12].

CP-ABE. In these schemes, the secret keys are associated with a set of attributes, while an access structure is embedded in the ciphertext [3, 4, 13].

In CP-ABE schemes, the encryptor can freely determine the access structure, while, in KP-ABE schemes, it is decided by the authority.

2.2 Multi-Authority Attribute-Based Encryption

In the work [2], Sahai and Waters left an open problem, namely how to construct an ABE scheme where the secret keys can be obtained from multiple

authorities so that users can reduce the trust on the central authority. Chase [9] answered this question affirmatively by proposing an MA-ABE scheme. The technical hurdle in designing an MA-ABE scheme is to resist the collusion attacks. To overcome this hurdle, GID was introduced to tie all the user's secret keys together. In [9], there is a central authority, and multiple authorities must interact to initialize the system.

Based on the distributed key generation (DKG) protocol [14] and the joint zero secret sharing (JZSS) protocol [15], Lin *et al.* [16] proposed an MA-ABE scheme where the cental authority is not required. To initialize the system, the multiple authorities must cooperatively execute the DKG protocol and the JZSS protocol twice and k times, respectively, where k is the degree of the polynomial selected by each authority. Each authority must maintain $k+2$ secret keys. This scheme is k-resilient, namely the scheme is secure if and only if the number of the colluding users is no more than k, and k must be fixed in the setup stage.

M*ü*ller *et al.* [17] proposed a distributed CP-ABE scheme which was proven to be secure in the generic group [3], instead of reducing to a complexity assumption. Furthermore, a central authority is required to generate the global key and issue secret keys to users.

Liu *et al.* [18] proposed a fully secure multi-authority CP-ABE scheme in the standard model. This scheme was derived from the CP-ABE scheme [7]. In this scheme, there are multiple central authorities and attribute authorities. The central authorities issue identity-related keys to users, while the attribute authorities issue attribute-related keys to users. Prior to possessing attribute keys from the attribute authorities, the user must obtain secret keys from the multiple central authorities. This MA-ABE scheme was designed in the composite order ($N = p_1p_2p_3$) bilinear group.

Lekwo and Waters [10] proposed a new MA-ABE scheme named decentralizing CP-ABE (DCP-ABE) scheme. This scheme improved the previous MA-ABE schemes that require collaborations among multiple authorities to conduct the system setup. In this scheme, no cooperation between the multiple authorities is required in the setup stage and the key generation stage, and there is no central authority. Notably, an authority in this scheme can join or leave the system freely without reinitializing the system. The scheme was constructed in the composite order ($N = p_1p_2p_3$) bilinear group, and achieves full (adaptive) security in the random oracle model. They also pointed out two methods to create a prime order group variant of their scheme. Nevertheless, the authorities can collect a user's attributes by tracing his GID.

Considering the privacy issues in MA-ABE schemes, Chase and Chow proposed [11] a new MA-ABE scheme which improved the previous scheme [9] and removed the need of a central authority. In previous MA-ABE schemes [9, 16], to obtain the corresponding secret keys, a user must submit his GID to each authority. So, multiple authorities can cooperate to collect the user's attributes by it. In [11], Chase and Chow provided an anonymous key issuing protocol for the GID where the 2-party secure computing technique is employed. As a result, a group of authorities cannot cooperate to pool the users attributes by tracing his

GID. However, the multiple authorities must collaborate to setup the system. Furthermore, each pair of authorities must execute the 2-party key exchange protocol to share the seeds of the selected pseudo random functions (PRFs) [19]. This scheme is $N-2$ tolerant, namely the scheme is secure if and only if the number of the corrupted authorities is no more than $N-2$, where N is the number of the authorities in the system. Although the authorities cannot know any information about the user's GID, they can know the user's attributes. Chase and Chow [11] also left an open challenging research problem on how to construct a privacy-preserving MA-ABE scheme without the need of cooperations among authorities.

Li [20] proposed a multi-authority CP-ABE (MACP-ABE) scheme with accountability, where the anonymous key issuing protocol [11] was employed. In this scheme, a user can be identified when he shared his secret keys with others. Notably, the multiple authorities must initialize the system interactively.

Recently, Han *et al.* [12] proposed a privacy-preserving decentralized KP-ABE (PPDKP-ABE) scheme. In this scheme, multiple authorities can work independently without any cooperation. Especially, the central authority is not required and a user can obtain secret keys from multiple authorities without releasing anything about his GID to them. Qian *et al.* [21] proposed a privacy-preserving decentralized CP-ABE (PPDCP-ABE) scheme which can support simple access structures. Nevertheless, similar to that in [11], the authorities in these schemes can know the user's attributes.

2.3 Anonymous Credential

In an anonymous credential system [22], an identity provider can issue a credential to a user, which includes the user's pseudonym and attributes. By using it, the user can prove in zero knowledge to a third party that he obtains a credential containing the given pseudonym and attributes without releasing any other information. In a multiple-show credential system [23], a credential can be demonstrated an arbitrary number of times, and cannot be linked to each other.

Therefore, in our construction, we assume that each user has obtained an anonymous credential including his GID and attributes. Then, he can prove in zero knowledge to the multiple authorities that he has a GID and holds the corresponding attributes using the anonymous credential technique.

3 Preliminaries

In this section, we introduce the preliminaries used throughout this paper.

In the remainder, by $\alpha \xleftarrow{\$} A$, we denote that α is selected from A randomly. Especially, $\alpha \xleftarrow{\$} A$ stands for that α is selected from A uniformly at random if A is a finite set. By $|A|$, we denote the cardinality of a finite set A. A function $\epsilon : \mathbb{Z} \to R$ is negligible if for any $z \in \mathbb{Z}$ there exists a k such that $\epsilon(x) < \frac{1}{x^z}$ when $x > k$. By $A(x) \to y$, we denote that y is computed by running the algorithm A with input x. $\mathcal{KG}(1^\kappa)$ denotes a secret-public key pair generator which takes as

input a security parameter 1^κ and outputs a secret-public key pair. We denote $\mathbb{Z}_p$ as a finite field with prime order p. Finally, by $R \xrightarrow{r} S$ and $R \xleftarrow{s} S$, we denote that the party R sends r to the party S and the party S sends s to the party R, respectively.

3.1 Complexity Assumption

Let $\mathbb{G}$ and $\mathbb{G}_\tau$ be two cyclic groups with prime order p, and g be a generator of $\mathbb{G}$. A map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_\tau$ is a bilinear group if the following properties can be satisfied:

1. Bilinearity. For all $a, b \in \mathbb{Z}_p$ and $u, v \in \mathbb{G}$, $e(u^a, v^b) = e(u^b, v^a) = e(u, v)^{ab}$.
2. Nondegeneracy. $e(g, g) \neq 1_\tau$ where 1_τ is the identity of the group $\mathbb{G}_\tau$.
3. Computability. For all $u, v \in \mathbb{G}$, there exists an efficient algorithm to compute $e(u, v)$.

Let $\mathcal{GG}(1^\kappa)$ be a bilinear group generator, which takes as input a security parameter 1^κ and outputs a bilinear group $(e, p, \mathbb{G}, \mathbb{G}_\tau)$ with prime order p and a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_\tau$.

Definition 1. (q-Strong Diffie-Hellman (q-SDH) Assumption [24]) *Let $x \xleftarrow{\$} \mathbb{Z}_p$, $\mathcal{GG}(1^\kappa) \to (e, p, \mathbb{G}, \mathbb{G}_\tau)$ and g be a generator of $\mathbb{G}$. Given a $(q+1)$-tuple $\overrightarrow{y} = (g, g^x, g^{x^2}, \cdots, g^{x^q})$, we say that the q-SDH assumption holds on $(e, p, \mathbb{G}, \mathbb{G}_\tau)$ if no probabilistic polynomial-time adversary $\mathcal{A}$ can output $(c, g^{\frac{1}{x+c}})$ with the advantage*

$$Adv_{\mathcal{A}} = \Pr[\mathcal{A}(\overrightarrow{y}) \to (c, g^{\frac{1}{x+c}})] \geq \epsilon(k)$$

where $c \in \mathbb{Z}_p$ and the probability is token over the random choices $x \xleftarrow{\$} \mathbb{Z}_p$ and the random bits consumed by $\mathcal{A}$.

Definition 2. (Decisional q-Parallel Bilinear Diffie-Hellman Exponent (q-PBDHE) Assumption [8]) *Let $a, s, b_1, \cdots, b_q \xleftarrow{\$} \mathbb{Z}_p$, $\mathcal{GG}(1^\kappa) \to (e, p, \mathbb{G}, \mathbb{G}_\tau)$ and g be a generator of $\mathbb{G}$. Given a tuple $\overrightarrow{y} =$*

$$g, g^s, g^a, \cdots, g^{(a^q)}, g^{(a^{q+2})}, \cdots, g^{(a^{2q})}$$
$$\forall_{1 \leq j \leq q} \quad g^{s \cdot b_j}, g^{\frac{a}{b_j}}, \cdots, g^{(\frac{a^q}{b_j})}, g^{(\frac{a^{q+2}}{b_j})}, \cdots, g^{(\frac{a^{2q}}{b_j})}$$
$$\forall_{1 \leq j,k \leq q, k \neq j} \quad g^{\frac{a \cdot s \cdot b_k}{b_j}}, \cdots, g^{(\frac{a^q \cdot s \cdot b_k}{b_j})},$$

we say that the decisional q-PBDHE assumption hold on $(e, p, \mathbb{G}, \mathbb{G}_\tau)$ if no probabilistic polynomial-time adversary $\mathcal{A}$ can distinguish $(\overrightarrow{y}, e(g, g)^{a^{q+1}s})$ from $(\overrightarrow{y}, R)$ with the advantage

$$Adv_{\mathcal{A}} = \left| \Pr[\mathcal{A}(\overrightarrow{y}, e(g, g)^{a^{q+1}s}) = 1] - \Pr[\mathcal{A}(\overrightarrow{y}, R) = 1] \right| \geq \epsilon(k),$$

where $R \xleftarrow{\$} \mathbb{G}_\tau$ and the probability is token over the random choices of $a, s, b_1, \cdots, b_q \xleftarrow{\$} \mathbb{Z}_p$ and the bits consumed by $\mathcal{A}$.

3.2 Building Blocks

In this paper, the following building blocks are adopted.

Definition 3. (Access Structure [25]) *Let $\mathcal{P} = (P_1, P_2, \cdots, P_n)$ be n parties. A collection $\mathbb{A} \subseteq 2^{\{P_1, P_2, \cdots, P_n\}}$ is monotonic if $B \in \mathbb{A}$ and $B \subseteq C$, then $C \in \mathbb{A}$. An access structure (respectively monotonic access structure) is a collection (respectively monotonic collection) $\mathbb{A}$ of the non-empty subset of $(P_1, P_2, \cdots, P_n)$, i.e., $\mathbb{A} \subseteq 2^{\{P_1, P_2, \cdots, P_n\}} \setminus \{\phi\}$. A set P is called an authorized set if $P \in \mathbb{A}$; otherwise P is an unauthorized set.*

Definition 4. (Linear Secret Sharing Schemes [25]) *A secret sharing scheme $\prod$ over a set of parties $\mathcal{P}$ is called linear (over $\mathbb{Z}_p$) if it satisfies the following properties:*

1. *The shares for each party form a vector over $\mathbb{Z}_p$.*
2. *For $\prod$, there is a matrix M with ℓ rows and n columns called the share-generating matrix. For $x = 1, 2, \cdots, \ell$, the ith row is labeled by a party $\rho(i)$ where $\rho : \{1, 2, \cdots, \ell\} \rightarrow \mathbb{Z}_p$. When we consider the vector $\overrightarrow{v} = (s, v_2, \cdots, v_n)$, where $s \in \mathbb{Z}_p$ is the secret to be shared and $v_2, \cdots, v_n \in \mathbb{Z}_p$ are randomly selected, then $M\overrightarrow{v}$ is the vector of the ℓ shares according to $\prod$. The share $M_i\overrightarrow{v}$ belongs to the party $\rho(i)$, where M_i is the ith row of M.*

Linear reconstruction property. Let S be an authorized set and $\mathcal{I} = \{i | \rho(i) \in S\}$. Then, there exist constants $\{\omega_i \in \mathbb{Z}_p\}_{i \in \mathcal{I}}$ such that, for any valid shares λ_i according to $\prod$, we have $\sum_{i \in \mathcal{I}} \omega_i \lambda_i = s$. The constants $\{\omega_i\}_{i \in \mathcal{I}}$ can be computed in polynomial time with the size of share-generating matrix M.

Commitment Schemes. A commitment scheme consists of the following algorithms.

$\mathsf{Setup}(1^\kappa) \rightarrow params$. This algorithm takes as input a security parameter 1^κ, and outputs the public parameters *params*.

$\mathsf{Commit}(params, m) \rightarrow (com, decom)$. This algorithm takes as input the public parameters *params* and a message m, and outputs a commitment *com* and a decommitment *decom*. *decom* can be used to decommit *com* to m.

$\mathsf{Decommit}(params, m, com, decom) \rightarrow \{0, 1\}$. This algorithm takes as input the public parameters *params*, the message m, the commitment *com* and the decommitment *decom*, and outputs 1 if *decom* can decommit *com* to m; otherwise, it outputs 0.

A commitment scheme should provide two properties: *hiding* and *binding*. The hiding property requires that the message m keeps unreleased until the user releases it later. The binding property requires that only the value *decom* can be used to decommit the commitment *com* to m.

In this paper, we use the Pedersen commitment scheme [26] which is a perfectly hiding commitment scheme and is based on the discrete logarithm assumption. This scheme works as follows. Let $\mathbb{G}$ be a cyclic group with prime

order p, and $g_0, g_1, \cdots, g_k$ be generators of $\mathbb{G}$. To commit a tuple of messages $m_1, m_2, \cdots, m_k$, the user selects $r \xleftarrow{\$} \mathbb{Z}_p$, and computes $R = g_0^r g_1^{m_1} g_2^{m_2} \cdots g_k^{m_k}$. Then, the user can use r to decommit the commitment R.

Proof of Knowledge. We use the notion introduced by Camenisch and Stadler [27] to prove statements about discrete logarithm. By

$$\mathrm{PoK}\left\{(\alpha, \beta, \gamma) : y = g^\alpha h^\beta \wedge \tilde{y} = \tilde{g}^\alpha \tilde{h}^\gamma\right\},$$

we denote a zero knowledge proof of knowledge of integers α, β and γ such that $y = g^\alpha h^\beta$ and $\tilde{y} = \tilde{g}^\alpha \tilde{h}^\gamma$ hold on the group $\mathbb{G} = \langle g \rangle = \langle h \rangle$ and $\tilde{\mathbb{G}} = \langle g \rangle = \langle h \rangle$, respectively. Conventionally, the values in the parenthesis denote the knowledge that is being proven, while the rest of the values are known by the verifier. There exists an extractor that can be used to rewind the knowledge from the successful prover.

Set-Membership Proof. Camenisch *et al.* [28] proposed a set membership proof scheme. This scheme works as follows. Let $\mathcal{GG}(1^\kappa) \to (e, p, \mathbb{G}, \mathbb{G}_\tau)$, and g, h be generators of $\mathbb{G}$.

1. The verifier picks up $x \xleftarrow{\$} \mathbb{Z}_p$, and computes $Y = g^x$ and $T_i = g^{\frac{1}{x+i}}$ for $i \in \Phi$, where $\Phi \subseteq \mathbb{Z}_p$ is a finite set. Then, it sends $\{Y, (T_i)_{i \in \Phi}\}$ to the prover.
2. To prove $\sigma \in \Phi$, the prover selects $v, s, t, r, k \xleftarrow{\$} \mathbb{Z}_p$, and computes $C = g^\sigma h^r$, $D = g^s h^k$, $V = g^{\frac{v}{x+\sigma}}$ and $A = e(V, g)^{-s} \cdot e(g, g)^t$. Then, it sends (C, D, V, A) to the verifier.
3. The verifier selects $c \xleftarrow{\$} \mathbb{Z}_p$, and sends it to the prover.
4. The prover computes $z_\sigma = s - c\sigma$, $z_r = k - cr$ and $z_v = t - cv$, and sends (z_σ, z_k, z_t) to the verifier.
5. The verifier verifies $D \stackrel{?}{=} C^c g^{z_\sigma} h^{z_r}$ and $A \stackrel{?}{=} e(Y, v)^c \cdot e(V, g)^{-z_\sigma} \cdot e(g, g)^{z_r}$.

Theorem 1. *This protocol is a zero-knowledge argument of set-membership proof for a set Φ if the $|\Phi|$-SDH assumption holds on the bilinear group $(e, p, \mathbb{G}, \mathbb{G}_\tau)$ [28].*

3.3 DCP-ABE: Decentralized Cipher-Policy Attribute-Based Encryption

A DCP-ABE scheme consists of the following five algorithms.

Global Setup$(1^\kappa) \to params$. The global setup algorithm takes as input a security parameter 1^κ, and outputs the public parameter *params*. Suppose that there are N authorities $\{\breve{A}_1, \breve{A}_2, \cdots, \breve{A}_N\}$, and each authority $\breve{A}_i$ monitors a set of attributes $\tilde{A}_i$. Each user U has a unique global identifier GID_U and holds a set of attributes $\tilde{U}$.

Authority Setup$(1^\kappa) \to (SK_i, PK_i)$. Each authority $\breve{A}_i$ takes as input the security parameter 1^κ, and runs the authority setup algorithm to generate its secret-public key pair (SK_i, PK_i), where $\mathcal{KG}(1^\kappa) \to (SK_i, PK_i)$.

$\mathsf{Encrypt}(params, \mathcal{M}, (M_i, \rho_i, PK_i)_{i\in\mathcal{I}}) \rightarrow CT$. The encryption algorithm takes as input the public parameter $params$, a message $\mathcal{M}$, a set of access structures $(M_i, \rho_i)_{i\in\mathcal{I}}$ and a set of public keys $(PK_i)_{i\in\mathcal{I}}$, and outputs the ciphertext CT.

$\mathsf{KeyGen}(params, SK_i, GID_U, \tilde{U} \bigcap \tilde{A}_i) \rightarrow SK_U^i$. Each authority $\breve{A}_i$ runs the key generation algorithm with inputs of the public parameter $params$, his secret key SK_i, a user's global identifier GID_U and a set of attributes $\tilde{U} \bigcap \tilde{A}_i$ to generate a secret key SK_U^i for U.

$\mathsf{Decrypt}(params, GID, (SK_U^i)_{i\in\mathcal{I}}, CT) \rightarrow \mathcal{M}$. The decryption algorithm takes as input the public parameter $params$, the user's globe identifier GID_U, the secret keys $(SK_U^i)_{i\in\mathcal{I}}$ and the ciphertext CT, and outputs the message $\mathcal{M}$.

Definition 5. *A decentralized cipher-policy attribute-based encryption is correct if*

$$\Pr\left[\begin{array}{l|l} \mathsf{Decrypt}(params, & \mathsf{Global\ Setup}(1^\kappa) \rightarrow params; \\ GID, (SK_U^i)_{i\in\mathcal{I}}, & \mathsf{Authority\ Setup}(1^\kappa) \rightarrow (SK_i, Pk_i); \\ CT) \rightarrow \mathcal{M} & \mathsf{Encrypt}(params, \mathcal{M}, (M_i, \rho_i, PK_i)_{i\in\mathcal{I}}) \rightarrow CT; \\ & \mathsf{KeyGen}(params, SK_i, GID_U, \tilde{U} \bigcap \tilde{A}_i) \rightarrow SK_U^i \end{array}\right] = 1$$

where the probability is token over the random bits consumed by all the algorithms in the scheme.

3.4 Security Model of Decentralized Cipher-Policy Attribute-Based Encryption

We use the following game to define the security model of DCP-ABE schemes, which is executed between a challenger and an adversary $\mathcal{A}$. This model is called selective-access structure model, and is similar to that introduced in [9, 11, 12, 10, 8].

Initialization. The adversary $\mathcal{A}$ submits a list of corrupted authorities $\mathfrak{A} = \{\breve{A}_i\}_{i\in\mathcal{I}}$ and a set of access structures $\mathbb{A} = \{M_i^*, \rho_i^*\}_{i\in\mathcal{I}^*}$, where $\mathcal{I} \subseteq \{1, 2, \cdots, N\}$ and $\mathcal{I}^* \subseteq \{1, 2, \cdots, N\}$. There should be at lease an access structure $(M^*, \rho^*) \in \mathbb{A}$ which cannot be satisfied by the attributes monitored by the authorities in $\mathfrak{A}$ and the attributes selected by $\mathcal{A}$ to query secrete keys.

Global Setup. The challenger runs the Global Setup algorithm to generate the public parameters $params$, and sends them to $\mathcal{A}$.

Authority Setup. There are two cases.

1. For the authority $\breve{A}_i \subseteq \mathfrak{A}$, the challenger runs the Authority Setup algorithm to generate the secret-public key pair (SK_i, PK_i), and sends them to $\mathcal{A}$.
2. For the authority $\breve{A}_i \nsubseteq \mathfrak{A}$, the challenger runs the Authority Setup algorithm to generate the secret-public key pair (SK_i, PK_i), and sends the public key PK_i to $\mathcal{A}$.

Phase 1. $\mathcal{A}$ can query secret key for a user U with an identifier GID_U and a set of attributes $\tilde{U}$. The challenger runs the KeyGen algorithm to generate a secret key SK_U, and sends it to $\mathcal{A}$. This query can be made adaptively and repeatedly.

Challenge. $\mathcal{A}$ submits two messages $\mathcal{M}_0$ and $\mathcal{M}_1$ with the same length. The challenger flips an unbiased coin with $\{0,1\}$, and obtains a bit $b \in \{0,1\}$. Then, the challenger runs $\mathsf{Encrypt}(parmas, \mathcal{M}_b, (M_i^*,\ \rho^*, PK_i)_{i\in\mathcal{I}^*})$ to generate the challenged ciphertext CT^*, and sends CT^* to $\mathcal{A}$.

Phase 2. Phase 1 is repeated.

Guess. $\mathcal{A}$ outputs his guess b' on b. $\mathcal{A}$ wins the game if $b' = b$.

Definition 6. (Selective-Access Structure Secure DCP-ABE (IND-sAS-CPA)) *A decentralized cipher-policy attribute-based encryption (DCP-ABE) scheme is* $(T, q, \epsilon(\kappa))$ *secure in the selective-access structure model if no probably polynomial-time adversary* $\mathcal{A}$ *making* q *secret key queries can win the above game with the advantage*

$$Adv_{\mathcal{A}}^{DCP-ABE} = \left|\Pr[b' = b] - \frac{1}{2}\right| > \epsilon(\kappa)$$

where the probability is token over all the bits consumed by the challenger and the adversary.

3.5 PPDCP-ABE: Privacy-Preserving Decentralized Cipher-Policy Attribute-Based Encryption

A PPDCP-ABE has the same algorithms Global Setup, Authority Setup, Encrypt and Decrypt with the DCP-ABE scheme. The main difference is that we replace the KeyGen algorithm with a privacy-preserving key generation algorithm PP-KeyGen. Considering privacy issues, the authorities cannot know both the user's identifier and attributes in PPDCP-ABE schemes. This is motivated by the blind IBE schemes [29, 30]. The PPKeyGen algorithm is formally defined as follows.

$\mathsf{PPKeyGen}(U(params, GID_U, \tilde{U}, PK_i, decom_i,\ (decom_{i,j})_{a_{i,j}\in\tilde{U}\bigcap\tilde{A}_i}) \leftrightarrow$
$\breve{A}_i(params, SK_i, PK_i, com_i,\ (com_{i,j})_{a_{i,j}\in\bigcap\tilde{A}_i})) \rightarrow (SK_U^i, \mathsf{empty})$. This is an interactive algorithm executed between a user U and an authority $\breve{A}_i$. U runs the commitment algorithm $\mathsf{Commit}(params, GID_U) \rightarrow (com_i, decom_i)$ and $\mathsf{Commit}(params, a_{i,j}) \rightarrow (com_{i,j}, decom_{i,j})$ for the attribute $a_{i,j} \in \tilde{U}\bigcap\tilde{A}_i$, and sends $(com_i, (com_{i,j})_{a_{i,j}\in\bigcap\tilde{A}_i})$ to the authority $\breve{A}_i$. Then, U and $\breve{A}_i$ take as input $(params, GID_U, \tilde{U}, PK_i, decom_i, (decom_{i,j})_{a_{i,j}\in\tilde{U}\bigcap\tilde{A}_i})$ and $(params, SK_i, PK_i,\ com_i, (com_{i,j})_{a_{i,j}\in\bigcap\tilde{A}_i})$, respectively. If $\mathsf{Decommit}(params, GID_U, com_i, dcom_i) = 1$ and $\mathsf{Decommit}(params, a_{i,j}, com_{i,j}, decom_{i,j}) = 1$, this algorithm outputs a secret key SK_U^i for U and an empty bit empty for $\breve{A}_i$; otherwise, it outputs $(\perp, \perp)$ to indicate that there are error messages.

3.6 Security Model of Privacy-Preserving Decentralized Cipher-Policy Attribute-Based Encryption

Now, we define the security of a PPDCP-ABE scheme, which informally is any IND-sAS-CPA-secure DCP-ABE scheme with a privacy-preserving key extract algorithm PPKeyGen that satisfies two properties: *leak-freeness* and *selective-failure blindness*. Leak-freeness requires that by executing the algorithm PPKeyGen with honest authorities, the malicious user cannot know anything which it cannot know by executing the algorithm KeyGen with the authorities. Selective-failure blindness requires that malicious authorities cannot know anything about the user's identifier and his attributes, and cause the PPKeyGen algorithm to selectively fail depending on the user's identifier and his attributes. These two properties can be formalized by using the following games.

Leak-Freeness. This game is defined by a real world experiment and an ideal world experiment.

Real World Experiment. Runs the Global Setup algorithm and Authority Setup algorithm. As many as the distinguisher $\mathcal{D}$ wants, the malicious user $\mathcal{U}$ chooses a global identifier $GID_{\mathcal{U}}$ and a set of attributes $\tilde{\mathcal{U}}$, and executes PPKeyGen($U(params, GID_{\mathcal{U}}, \tilde{\mathcal{U}}, PK_i, decom_i, (decom_{i,j})_{a_{i,j} \in \tilde{\mathcal{U}} \bigcap \tilde{A}_i}) \leftrightarrow \breve{A}_i(params, SK_i, PK_i, com_i, (com_{i,j})_{a_{i,j} \in \bigcap \tilde{\mathcal{U}} \bigcap \tilde{A}_i})) \rightarrow (SK^i_{\mathcal{U}}, \text{empty})$ with $\breve{A}_i$.

Ideal World Experiment. Runs the Global Setup algorithm and Authority Setup algorithm. As many as the distinguisher $\mathcal{D}$ wants, the malicious user $\bar{U}$ chooses a global identifier $GID_{\bar{U}}$ and a set of attributes $\tilde{\bar{U}}$, and requires a trusted party to obtain the output of KeyGen$(params, SK_i, GID_{\bar{U}}, \tilde{\bar{U}} \bigcap \tilde{A}_i) \rightarrow SK^i_{\bar{U}}$.

Definition 7. *An algorithm* PPKeyGen$(U \leftrightarrow \breve{A}_i)$ *associated with a DCP-ABE scheme* $\prod =$ (GlobalSetup, AuthoritySetup, Encrypt, KeyGen, Decrypt) *is leak-free if for all efficient adversary* $\mathcal{U}$*, there exists a simulator* $\bar{U}$ *such that, for the security parameter* 1^κ*, no distinguisher* $\mathcal{D}$ *can distinguish whether* $\mathcal{U}$ *is playing in the real world experiment or in the ideal world experiment with non-negligible advantage.*

Selective-Failure Blindness. This game is formalized as follows.

1. The malicious authority $\mathcal{A}_i$ outputs his public key PK_i and two pairs of globe identifiers and attribute sets $(GID_{U_0}, \tilde{U}_0)$ and $(GID_{U_1}, \tilde{U}_1)$.
2. A random bit $b \in \{0, 1\}$ is selected.
3. $\mathcal{A}_i$ is given comments

$$\left\{com_b, (com_{i,j})_{a_{i,j} \in \tilde{U}_b \bigcap \tilde{\mathcal{A}}_i}\right\} \text{ and } \left\{com_{1-b}, (com_{i,j})_{a_{i,j} \in \tilde{U}_{1-b} \bigcap \tilde{\mathcal{A}}_i}\right\},$$

   and can black-box access oracles $U(params, GID_{U_b}, \tilde{U}_b, PK_i, decom_b, (decom_{i,j})_{a_{i,j} \in \tilde{U}_b \bigcap \tilde{\mathcal{A}}_i})$ and $U(params, GID_{U_{1-b}}, \tilde{U}_{1-b}, PK_i, decom_{1-b}, (decom_{i,j})_{a_{i,j} \in \tilde{U}_{1-b} \bigcap \tilde{\mathcal{A}}_i})$.
4. The algorithm U outputs the secret keys $SK^i_{U_b}$ and $SK^i_{U_{1-b}}$, respectively.

5. If $SK^i_{U_b} \neq \perp$ and $SK^i_{U_{1-b}} \neq \perp$, $\mathcal{A}_i$ is given $(SK^i_{U_b}, SK^i_{U_{1-b}})$; if $SK^i_{U_b} \neq \perp$ and $SK^i_{U_{1-b}} = \perp$, $\mathcal{A}_i$ is given $(\epsilon, \perp)$; if $SK^i_{U_b} = \perp$ and $SK^i_{U_{1-b}} \neq \perp$, $\mathcal{A}_i$ is given $(\perp, \epsilon)$; if $SK^i_{U_b} = \perp$ and $SK^i_{U_{1-b}} = \perp$, $\mathcal{A}_i$ is given $(\perp, \perp)$.
6. $\mathcal{A}_i$ outputs his guess b' on b. $\mathcal{A}_i$ wins the game if $b' = b$.

Definition 8. *An algorithm* $\mathsf{PPKeyGen}(U \leftrightarrow \breve{A}_i)$ *associated to a DCP-ABE scheme* $\prod = (\mathsf{Global\ Setup}, \mathsf{Authority\ Setup}, \mathsf{Encrypt}, \mathsf{KeyGen}, \mathsf{Decrypt})$ *is selective-failure blind if no probably polynomial-time adversary* $\mathcal{A}_i$ *can win the above game with the advantage* $Adv^{SFB}_{\mathcal{A}_i} = \left|\Pr[b' = b] - \frac{1}{2}\right| > \epsilon(\kappa)$*, where the probability is token over the bits consumed by all the algorithms and the adversary.*

Definition 9. *A privacy-preserving decentralized cipher-policy attribute-based encryption (PPDCP-ABE) scheme* $\widetilde{\prod} = (\mathsf{Global\ Setup}, \mathsf{Authority\ Setup}, \mathsf{Encrypt}, \mathsf{PPKeyGen}, \mathsf{Decrypt})$ *is secure if and only if the following conditions are satisfied:*

1. $\prod = (\mathsf{Global\ Setup}, \mathsf{Authority\ Setup}, \mathsf{Encrypt}, \mathsf{KeyGen}, \mathsf{Decrypt})$ *is a secure DCP-ABE in the selective-access structures model;*
2. *the* PPKeyGen *algorithm is both leak-free and selective-failure blind.*

4 Our Constructions

In this session, we first construct a DCP-ABE scheme, and then propose a privacy-preserving key extract protocol for it.

4.1 DCP-ABE: Decentralized Cipher-Policy Attribute-Based Encryption

Overview. Suppose that there are N authorities $\{\breve{A}_1, \breve{A}_2, \cdots, \breve{A}_N\}$ in the scheme, and each authority $\breve{A}_i$ monitors a set of attributes $\tilde{A}_i$ for $i = 1, 2, \cdots, N$. $\breve{A}_i$ generates his secret-public key pair $\mathcal{KG}(1^\kappa) \rightarrow (SK_i, PK_i)$. For each attribute $a_{i,j} \in \tilde{A}_i$, $\breve{A}_i$ chooses a random number $z_{i,j} \xleftarrow{\$} \mathbb{Z}_p$. Then, the public key is computed as $Z_{i,j} = g^{z_{i,j}}$ and the authentication tag is computed as $T_{i,j} = h^{z_{i,j}} g^{\frac{1}{\gamma_i + a_{i,j}}}$ where γ_i is the partial secret key of $\breve{A}_i$. $T_{i,j}$ can be used to convince $\breve{A}_i$ that the attribute $a_{i,j}$ is monitored by him without revealing it.

To encrypt a message $\mathcal{M}$ under the attributes monitored by the authorities $\{\breve{A}_j\}_{j \in \mathcal{I}}$, the encryptor selects a random number $s_j \xleftarrow{\$} \mathbb{Z}_p$ and an access structure (M_j, ρ_j) for each $\breve{A}_j$. Then, it splits s_j into shares $\lambda_{j,i}$ according to the LSSS technique. Finally the message $\mathcal{M}$ is blinded with $\prod_{j \in \mathcal{I}} e(g, g)^{\alpha_j s_j}$.

To resist the collusion attack, when generating a secret key for a user U with GID μ and a set of attributes $\tilde{U}$, $\breve{A}_i$ chooses two random numbers $(t_{U,i}, w_{U,i}) \xleftarrow{\$} \mathbb{Z}_p$. Specifically, $t_{U,i}$ is used to tie the user's attribute keys to his GID by computing $\mathfrak{g}^{t_{U,i}} \mathfrak{g}^{\frac{\beta_i + \mu}{t_{U,i}}}$ where β_i is the partial secret key of $\breve{A}_i$, and

$w_{U,i}$ is used to randomize the public keys by computing $(F_x = Z_x^{w_{U,i}})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}$. Then, $\breve{A}_i$ can generate a secret key for U by using his secret key and $(t_{U,i}, w_{U,i})$.

To decrypt a ciphertext, each $e(g,g)^{\alpha_j s_j}$ must be recovered. If the attributes in $\tilde{U}$ satisfy the access structures $(M_j, \rho_j)_{j \in \mathcal{I}}$, the user can use his secret keys and the corresponding ciphertexte elements to reconstruct $e(g,g)^{\alpha_j s_j}$, and obtain $\mathcal{M}$.

Our DCP-ABE scheme is described in Fig.1.

Correctness. Our scheme in Fig. 1 is correct as we have

$$\prod_{j \in \mathcal{I}} e(K_j, X_j) = \prod_{j \in \mathcal{I}} e(g^{\alpha_j} g^{x_j w_{U,j}} \mathfrak{g}^{t_{U,j}} \mathfrak{g}^{\frac{\beta_j + \mu}{t_{U,j}}}, g^{s_j}) =$$
$$\prod_{j \in \mathcal{I}} e(g,g)^{\alpha_j s_j} \cdot e(g,g)^{x_j w_{U,j} s_j} \cdot e(g, \mathfrak{g})^{t_{U,j} s_j} \cdot e(g, \mathfrak{g})^{\frac{\beta_j s_j}{t_{U,j}}} \cdot e(g, \mathfrak{g})^{\frac{\mu s_j}{t_{U,j}}},$$

$$\prod_{j \in \mathcal{I}} e(R_j, E_j) \cdot e(R_j, Y_j)^{\mu} = \prod_{j \in \mathcal{I}} e(g^{\frac{1}{t_{U,j}}}, B_j^{s_j}) \cdot e(g^{\frac{1}{t_{U,j}}}, \mathfrak{g}^{s_j})^{\mu} =$$
$$\prod_{j \in \mathcal{I}} e(g^{\frac{1}{t_{U,j}}}, \mathfrak{g}^{\beta_j s_j}) \cdot e(g^{\frac{1}{t_{U,j}}}, \mathfrak{g}^{s_j})^{\mu} = \prod_{j \in \mathcal{I}} e(g, \mathfrak{g})^{\frac{\beta_j s_j}{t_{U,j}}} \cdot e(g, \mathfrak{g})^{\frac{\mu s_j}{t_{U,j}}},$$

$$\prod_{j \in \mathcal{I}} e(L_j, X_j) = e(g,g)^{t_{U,j} s_j},$$

$$\prod_{j \in \mathcal{I}} \prod_{i=1}^{\ell_j} \left(e(C_{j,i}, P_j) \cdot e(D_{j,i}, F_{\rho_j(i)}) \right)^{\omega_{j,i}} =$$
$$\prod_{j \in \mathcal{I}} \prod_{i=1}^{\ell_j} \left(e(g^{x_j \lambda_{j,i}} Z_{\rho_j(i)}^{-r_{j,i}}, g^{w_{U,j}}) \cdot e(g^{r_{j,i}}, Z_{\rho_j(i)}^{w_{U,j}}) \right)^{\omega_{j,i}} =$$
$$\prod_{j \in \mathcal{I}} e(g,g)^{x_j w_{U,j} \sum_{i=1}^{\ell_j} \omega_{j,i} \lambda_{j,i}} = \prod_{j \in \mathcal{I}} e(g,g)^{x_j w_{U,j} s_j}.$$

Therefore,

$$\frac{C_0 \cdot \prod_{j \in \mathcal{I}} e(L_j, X_j) \cdot e(R_j, E_j) \cdot e(R_j, Y_j)^{\mu}}{\prod_{j \in \mathcal{I}} e(K_j, X_j)} \cdot \prod_{j \in \mathcal{I}} \prod_{i=1}^{\ell_j} \left(e(C_{j,i}, P_j) \cdot e(D_{j,i}, F_{\rho_j(i)}) \right)^{\omega_{j,i}}$$
$$= \mathcal{M}.$$

Theorem 2. *Our decentralized cipher-policy attribute-based encryption (DCP-ABE) is* $(T, q, \epsilon(k))$ *secure in the selective-access structure model if the* $(T', \epsilon'(k))$*-decisional q-PBDHE assumption holds on* $(e, p, \mathbb{G}, \mathbb{G}_\tau)$*, where* $T' = T + \mathcal{O}(T)$ *and* $\epsilon'(\kappa) = \frac{1}{2}\epsilon(\kappa)$.

The proof of this theorem is referred to the full version of this paper [31].

4.2 Privacy-Preserving Key Extract Protocol

In this session, we propose a privacy-preserving key extract protocol for the DCP-ABE scheme described in Fig. 1.

Overview. In Fig. 1, to generate a secret key for a user U, the authority $\breve{A}_i$ selects two random numbers $(t_{U,i}, w_{U,i})$, and uses them to tie the user's secret keys to his GID. If $\breve{A}_i$ records $(t_{U,i}, w_{U,i})$, he can compute $\mathfrak{g}^{\mu} = (\frac{K_i}{g^{\alpha_i} g^{x_i w_{U,i}} \mathfrak{g}^{t_{U,i}}})^{t_{U,i}} \mathfrak{g}^{-\beta_i}$ and $(Z_x = F_x^{\frac{1}{w_{U,i}}})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}$. Hence, he can know the user's GID and attributes. Therefore, in order to protect the privacy of the user's GID and attributes, $(t_{U,i}, w_{U,i})$ should be computed using the 2-party secure computing technique.

Global Setup. This algorithm takes as input a security parameter 1^κ, and outputs a bilinear group $\mathcal{GG}(1^\kappa) \to (e, p, \mathbb{G}, \mathbb{G}_\tau)$. Let g, h and $\mathfrak{g}$ be generators of the group $\mathbb{G}$. Suppose that there are N authorities $\{\breve{A}_1, \breve{A}_2, \cdots, \breve{A}_N\}$, and $\breve{A}_i$ monitors a set of attributes $\tilde{A}_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,q_i}\}$ where $a_{i,j} \in \mathbb{Z}_p$ for $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, q_i$. The public parameters are $PP = (g, h, \mathfrak{g}, e, p, \mathbb{G}, \mathbb{G}_\tau)$.

Authorities Setup. Each authority $\breve{A}_i$ selects $\alpha_i, x_i, \beta_i, \gamma_i \xleftarrow{\$} \mathbb{Z}_p$, and computes $H_i = e(g,g)^{\alpha_i}$, $A_i = g^{x_i}$, $B_i = \mathfrak{g}^{\beta_i}$, $\Gamma_i^1 = g^{\gamma_i}$ and $\Gamma_i^2 = h^{\gamma_i}$, where $i = 1, 2, \cdots, N$. For each attribute $a_{i,j} \in \tilde{A}_i$, $\breve{A}$ chooses $z_{i,j} \xleftarrow{\$} \mathbb{Z}_p$, and computes $Z_{i,j} = g^{z_{i,j}}$ and $T_{i,j} = h^{z_{i,j}} g^{\frac{1}{\gamma_i + a_{i,j}}}$. Then, $\breve{A}$ publishes the public key $PK_i = \left\{H_i, A_i, B_i, (\Gamma_i^1, \Gamma_i^2), (T_{i,j}, Z_{i,j})_{a_{i,j} \in \tilde{A}_i}\right\}$, and keeps the master secrete key as $SK_i = (\alpha_i, a_i, \beta_i, \gamma_i, (z_{i,j})_{a_{i,j} \in \tilde{A}_i})$.

Encryption. To encrypt a message $\mathcal{M} \in \mathbb{G}_\tau$, this algorithm works as follows. Let $\mathcal{I}$ be a set which consists of the indexes of the authorities whose attributes are selected to encrypt $\mathcal{M}$. For each $j \in \mathcal{I}$, this algorithm first selects an access structures (M_j, ρ_j) and a vector $\overrightarrow{v_j} = (s_j, v_{j,2}, \cdots, v_{j,n_j})$, where $s_j, v_{j,2}, \cdots, v_{j,n_j} \xleftarrow{\$} \mathbb{Z}_p$ and M_j is an $\ell_j \times n_j$ matrix. Then, it computes $\lambda_{j,i} = M_j^i \overrightarrow{v}_j$, where M_j^i is the corresponding ith row of M_j. Finally, it selects $r_{j,1}, r_{j,2}, \cdots, r_{j,\ell_j} \xleftarrow{\$} \mathbb{Z}_p$, and computes

$$C_0 = \mathcal{M} \cdot \prod_{j \in \mathcal{I}} e(g,g)^{\alpha_j s_j},\ \{X_j = g^{s_j},\ Y_j = \mathfrak{g}^{s_j},\ E_j = B_j^{s_j}\}_{j \in \mathcal{I}}$$

$$\left((C_{j,1} = g^{x_j \lambda_{j,1}} Z_{\rho_j(1)}^{-r_{j,1}},\ D_{j,1} = g^{r_{j,1}}),\ \cdots,\ (C_{j,\ell_j} = g^{x_j \lambda_{j,\ell_j}} Z_{\rho_j(\ell_j)}^{-r_{j,\ell_j}},\ D_{j,\ell_j} = g^{r_{j,\ell_j}})\right)_{j \in \mathcal{I}}$$

The ciphertext is $CT = \left\{C_0,\ \left(X_j,\ Y_j,\ E_j,\ (C_{j,1},\ D_{j,1}),\ \cdots,\ (C_{j,\ell_j},\ D_{j,\ell_j})\right)_{j \in \mathcal{I}}\right\}$.

KeyGen. To generate secret keys for a user U with GID μ and a set of attributes $\tilde{U} \bigcap \tilde{A}_i$, $\breve{A}_i$ selects $t_{U,i}, w_{U,i} \xleftarrow{\$} \mathbb{Z}_p$, and computes $K_i = g^{\alpha_i} g^{x_i w_{U,i}} \mathfrak{g}^{t_{U,i}} \mathfrak{g}^{\frac{\beta_i + \mu}{t_{U,i}}}$, $P_i = g^{w_{U,i}}$, $L_i = g^{t_{U,i}}$, $L_i' = h^{t_{U,i}}$, $R_i = g^{\frac{1}{t_{U,i}}}$, $R_i' = h^{\frac{1}{t_{U,i}}}$ and $(F_x = Z_x^{w_{U,i}})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}$. The secret keys for U are $SK_U^i = \left\{K_i, P_i, L_i, L_i', R_i, R_i', (F_x)_{a_x \in \tilde{U} \bigcap \tilde{A}_i}\right\}$.

Decryption. To decrypt a ciphertext CT, this algorithm computes

$$\frac{C_0 \cdot \prod_{j \in \mathcal{I}} e(L_j, X_j) \cdot e(R_j, E_j) \cdot e(R_j, Y_j)^\mu \cdot \prod_{j \in \mathcal{I}} \prod_{i=1}^{\ell_j} \left(e(C_{j,i}, P_j) \cdot e(D_{j,i}, F_{\rho_j(i)})\right)^{\omega_{j,i}}}{\prod_{j \in \mathcal{I}} e(K_j, X_j)}$$

$$= \mathcal{M}$$

where $\{\omega_{j,i} \in \mathbb{Z}_p\}_{i=1}^{\ell_j}$ are a set of constants such that $\sum_{i=1}^{\ell_j} \omega_{j,i} \lambda_{j,i} = s_j$ if $\{\lambda_{j,i}\}_{i=1}^{\ell_j}$ are valid shares of the secret value s_j according to the access structure (M_j, ρ_j).

Fig. 1. Decentralized Cipher-Policy Attribute-based Encryption

$U(PP, PK_i, \mu, a_x \in \tilde{U} \bigcap \tilde{A}_i)$ | $\breve{A}_i(PP, PK_i, SK_i)$

1. Selects $k_1, k_2, d_1, d_2 \xleftarrow{\$} \mathbb{Z}_p$ and sets $d_u = d_1 d_2$.
Computes
$\Theta_1 = A_i^{d_1}$, $\Theta_2 = g^{d_u}$,
$\Theta_3 = h^{k_1}\mathfrak{g}^{\mu}$, $\Theta_4 = \Theta_3^{k_2}$,
$\Theta_5 = B_i^{k_2}$, $\Theta_6 = \mathfrak{g}^{\frac{1}{k_2}}$,
$(\Psi_x^1 = T_x^{d_u}, \Psi_x^2 = Z_x^{d_u})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}$
and $\Sigma_U = \mathsf{PoK}\{(k_1, k_2, d_1, d_u, \mu,$
$(a_x \in \tilde{U} \bigcap \tilde{A}_i)) : \Theta_1 = A_i^{d_1} \wedge$
$\Theta_2 = g^{d_u} \wedge \Theta_3 = h^{k_1}\mathfrak{g}^{\mu}, \wedge$
$\Theta_4 = \Theta_3^{k_2} \wedge \Theta_5 = B_i^{k_2} \wedge$
$e(\Theta_5, \Theta_6) = e(B_i, \mathfrak{g}) \wedge$
$(\wedge \frac{e(\Gamma_i^1, \Psi_x^1)}{e(\Gamma_i^2, \Psi_x^2)} = e(g, \Psi_x^1)^{-a_x} \cdot$
$\wedge e(h, \Psi_x^2)^{a_x} \cdot e(g, g)^{d_u})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}\}$

$\xrightarrow[\Theta_5, \Psi_x^1, \Psi_x^2, \Sigma_U]{\Theta_1, \Theta_2, \Theta_3, \Theta_4}$

2. Selects $c_u, e_u \xleftarrow{\$} \mathbb{Z}_p$ and computes $\Upsilon_1 = g^{c_u}, \Upsilon_2 = g^{\frac{1}{c_u}}$,
$\Upsilon_3 = h^{c_u}, \Upsilon_4 = h^{\frac{1}{c_u}}, \Upsilon_5 = g^{e_u}$,
$K_i' = g^{\alpha_i}\Theta_1^{e_u}\Theta_6^{c_u}(\Theta_4\Theta_5)^{\frac{1}{c_u}}$,
$(\Phi_x = (\Psi_x^2)^{e_u})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}$ and
$\Sigma_{A_i} = \mathsf{PoK}\{(\alpha_i, c_u, e_u) :$
$e(\Upsilon_1, \Upsilon_2) = e(g, g) \wedge \Upsilon_1 = g^{c_u} \wedge$
$\Upsilon_2 = g^{\frac{1}{c_u}} \wedge \Upsilon_3 = h^{c_u} \wedge \Upsilon_4 = h^{\frac{1}{c_u}}$
$e(\Upsilon_3, \Upsilon_4) = e(h, h) \wedge \Upsilon_5 = g^{e_u} \wedge$
$K_i' = g^{\alpha_i}\Theta_1^{e_u}\Theta_6^{c_u}(\Theta_4\Theta_5)^{\frac{1}{c_u}}$
$\wedge(\wedge(\Phi_x = (\Psi_x^2)^{e_u})_{a_x \in \tilde{U} \bigcap \tilde{A}_i}\}$.

3. Computes $K_i = \frac{K_i'}{\Upsilon_4^{k_1 k_2}}$, $P_i = \Upsilon_5^{d_1}$,

$\xleftarrow[\Upsilon_5, K_i', \Phi_x, \Sigma_{A_i}]{\Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4}$

$L_i = \Upsilon_1^{\frac{1}{k_2}}$, $R_i = \Upsilon_2^{k_2}$, $R_i' = \Upsilon_4^{k_2}$ and
$\left(F_x = \Phi_x^{\frac{1}{d_2}}\right)_{a_x \in \tilde{U} \bigcap \tilde{A}_i}$

Fig. 2. PPKeyGen: Privacy-Preserving Key Generation Protocol

First, U chooses $(k_1, k_2, d_1, d_2) \xleftarrow{\$} \mathbb{Z}_p$. It uses (k_1, k_2) to commit his GID and (d_1, d_2) to commit his attributes and the corresponding authentication tags. Then, U proves in zero knowledge to $\breve{A}_i$ that he knows the GID, and the attributes for which he is obtaining secret keys are monitored by $\breve{A}_i$. $\breve{A}_i$ checks the proof. If it fails, $\breve{A}_i$ aborts. Otherwise, $\breve{A}_i$ chooses $(c_u, e_u) \xleftarrow{\$} \mathbb{Z}_p$ and generates a secret key for U by using his secret key, the elements from U and (c_u, e_u). Furthermore, $\breve{A}_i$ proves in zero knowledge that he knows the secret key and (c_u, e_u); Finally, U can compute his real secret key by (k_1, k_2, d_1, d_2) and the elements from $\breve{A}_i$.

Actually, by executing the 2-party secure computing protocol, U and $\breve{A}_i$ cooperatively compute $w_{U,i} = e_u d_1$ and $t_{U,i} = \frac{c_u}{k_2}$, where (d_1, k_2) are from U and (c_u, e_u) are from $\breve{A}_i$. Therefore, from the view of $\breve{A}_i$, the secret key computed by U is indistinguishable from the random elements in $\mathbb{G}$.

The privacy-preserving key extract protocol is described in Fig. 2.

Correctness. Let $w = d_1 e_u$ and $t = \frac{c_u}{k_2}$. The secret keys generated in Fig. 2 are correct as the following equations hold.

$$K_i = \frac{K_i' \Upsilon^{\frac{1}{k_2}}}{\Upsilon_4^{k_1 k_2}} = \frac{g^{\alpha_i} \Theta_1^{e_u} (\Theta_4 \Theta_5)^{\frac{1}{c_u}} \mathfrak{g}^{\frac{c_u}{k_2}}}{\Upsilon_4^{k_1 k_2}} = \frac{g^{\alpha_i} A_i^{d_1 e_u} ((\mathfrak{h}^{k_1} \mathfrak{g}^{\mu})^{k_2} B_i^{k_2})^{\frac{1}{c_u}} \mathfrak{g}^{\frac{c_u}{k_2}}}{\mathfrak{h}^{\frac{k_1 k_2}{c_u}}}$$

$$= \frac{g^{\alpha_i} g^{x_i d_1 e_u} \mathfrak{h}^{\frac{k_1 k_2}{c_u}} \mathfrak{g}^{\frac{k_2(\beta_i+\mu)}{c_u}} \mathfrak{g}^{\frac{c_u}{k_2}}}{\mathfrak{h}^{\frac{k_1 k_2}{c_u}}} = g^{\alpha_i} g^{x_i w} \mathfrak{g}^{t} \mathfrak{g}^{\frac{\beta_i+\mu}{t}},$$

$$P_i = \Upsilon_6^{d_1} = g^{d_1 e_u} = g^w,\ L_i = \Upsilon_1^{\frac{1}{k_2}} = g^{\frac{c_u}{k_2}} = g^t,$$

$$R_i = \Upsilon_2^{k_2} = g^{\frac{k_2}{c_u}} = g^{\frac{1}{t}},\ R_i' = \Upsilon_4^{k_2} = h^{\frac{k_2}{c_u}} = h^{\frac{1}{t}}$$

and

$$F_x = \Phi_x^{\frac{1}{d_2}} = (\Psi_x^2)^{\frac{e_u}{d_2}} = Z_x^{\frac{d_u e_u}{d_2}} = Z_x^{d_1 e_u} = Z_x^w.$$

Theorem 3. *The privacy-preserving key extract protocol in Fig. 2 is both leak-free and selective-failure blind under the q-SDH assumption, where* $q = max\{q_1, q_2, \cdots, q_N\}$.

The proof of this theorem is referred to the full version of this paper [31].

By **Theorem** 2 and **Theorem** 3, we have the following theorem.

Theorem 4. *Our privacy-preserving decentralized cipher-policy attribute-based encryption (PPDCP-ABE) scheme* $\prod = (\mathsf{Global\ Setup}, \mathsf{Authority\ Setup}, \mathsf{Encrypt}, \mathsf{PPKeyGen}, \mathsf{Decrypt})$ *is secure in the selective-access structure model under the decisional q-PBDHE assumption and q-SDH assumption.*

5 Conclusion

Decentralized ABE scheme is more efficient and flexible encryption system as it dose not require a central authority nor the cooperation among multiple authorities. Considering to reduce the trust on the authorities, some privacy-preserving MA-ABE schemes have been proposed. However, in these schemes, only the privacy of the GID was considered. In this paper, we proposed a PPDCP-ABE scheme where both the privacy of the GID and the attributes are concerned. Especially, the user can convince the authorities that the attributes for which he is obtaining secret keys are monitored by them. Therefore, our scheme provides a perfect solution for the privacy issues in MA-ABE schemes.

Acknowledgement. We would like to thank the anonymous reviewers for useful comments. The first author was partially supported by National Natural Science Foundation of China (Grant No. 61300213), National Center for International Joint Research on E-Business Information Processing (Grant No. 2013B01035) and A Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions(PAPD). The second author was partially supported by Australia Research Council Discovery Project (DP130101383).

References

1. Bichsel, P., Camenisch, J., Groβ, T., Shoup, V.: Anonymous credentials on a standard java card. In: Al-Shaer, E., Jha, S., Keromytis, A.D. (eds.) CCS 2009, pp. 600–610. ACM (2009)
2. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (2005)
3. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: IEEE S& P 2007, pp. 321–334. IEEE (2007)
4. Cheung, L., Newport, C.: Provably secure ciphertext policy ABE. In: Ning, P., di Vimercati, S.D.C., Syverson, P.F. (eds.) CCS 2007, pp. 456–465. ACM (2007)
5. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Juels, A., Wright, R.N., di Vimercati, S.D.C. (eds.) CCS 2006, pp. 89–98. ACM (2006)
6. Ostrovsky, R., Sahai, A., Waters, B.: Attribute-based encryption with non-monotonic access structures. In: Ning, P., di Vimercati, S.D.C., Syverson, P.F. (eds.) CCS 2007, pp. 195–203. ACM (2007)
7. Lewko, A., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 62–91. Springer, Heidelberg (2010)
8. Waters, B.: Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011. LNCS, vol. 6571, pp. 53–70. Springer, Heidelberg (2011)
9. Chase, M.: Multi-authority attribute based encryption. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 515–534. Springer, Heidelberg (2007)
10. Lewko, A., Waters, B.: Decentralizing attribute-based encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 568–588. Springer, Heidelberg (2011)
11. Chase, M., Chow, S.S.: Improving privacy and security in multi-authority attribute-based encryption. In: Al-Shaer, E., Jha, S., Keromytis, A.D. (eds.) CCS 2009, pp. 121–130. ACM (2009)
12. Han, J., Susilo, W., Mu, Y., Yan, J.: Privacy-preserving decentralized key-policy attribute-based encryption. IEEE Transactions on Parallel and Distributed Systems 23(11), 2150–2162 (2012)
13. Herranz, J., Laguillaumie, F., Ràfols, C.: Constant size ciphertexts in threshold attribute-based encryption. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 19–34. Springer, Heidelberg (2010)
14. Gennaro, R., Jarecki, S., Krawczyk, H., Rabin, T.: Secure distributed key generation for discrete-log based cryptosystems. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 295–310. Springer, Heidelberg (1999)
15. Gennaro, R., Jarecki, S., Krawczyk, H., Rabin, T.: Robust threshold dss signatures. In: Maurer, U.M. (ed.) Advances in Cryptology - EUROCRYPT 1996. LNCS, vol. 1070, pp. 354–371. Springer, Heidelberg (1996)
16. Lin, H., Cao, Z., Liang, X., Shao, J.: Secure threshold multi authority attribute based encryption without a central authority. In: Chowdhury, D.R., Rijmen, V., Das, A. (eds.) INDOCRYPT 2008. LNCS, vol. 5365, pp. 426–436. Springer, Heidelberg (2008)
17. Müller, S., Katzenbeisser, S., Eckert, C.: Distributed attribute-based encryption. In: Lee, P.J., Cheon, J.H. (eds.) ICISC 2008. LNCS, vol. 5461, pp. 20–36. Springer, Heidelberg (2009)

18. Liu, Z., Cao, Z., Huang, Q., Wong, D.S., Yuen, T.H.: Fully secure multi-authority ciphertext-policy attribute-based encryption without random oracles. In: Atluri, V., Diaz, C. (eds.) ESORICS 2011. LNCS, vol. 6879, pp. 278–297. Springer, Heidelberg (2011)
19. Naor, M., Pinkas, B., Reingold, O.: Distributed pseudo-random functions and KDCs. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 327–346. Springer, Heidelberg (1999)
20. Li, J., Huang, Q., Chen, X., Chow, S.S.M., Wong, D.S., Xie, D.: Multi-authority ciphertext-policy attribute-based encryption with accountability. In: Cheung, B.S.N., Hui, L.C.K., Sandhu, R.S., Wong, D.S. (eds.) ASIACCS 2011, pp. 386–390. ACM (2011)
21. Qian, H., Li, J., Zhang, Y.: Privacy-preserving decentralized ciphertext-policy attribute-based encryption with fully hidden access structure. In: Qing, S., Zhou, J., Liu, D. (eds.) ICICS 2013. LNCS, vol. 8233, pp. 363–372. Springer, Heidelberg (2013)
22. Camenisch, J., Lysyanskaya, A.: An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 93–118. Springer, Heidelberg (2001)
23. Persiano, G., Visconti, I.: An efficient and usable multi-show non-transferable anonymous credential system. In: Juels, A. (ed.) FC 2004. LNCS, vol. 3110, pp. 196–211. Springer, Heidelberg (2004)
24. Boneh, D., Boyen, X.: Short signatures without random oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 56–73. Springer, Heidelberg (2004)
25. Beime, A.: Secure Schemes for Secret Sharing and Key Distribution. Phd thesis, Israel Institute of Technology, Technion, Haifa, Israel (1996)
26. Pedersen, T.P.: Non-interactive and information-theoretic secure verifiable secret sharing. In: Feigenbaum, J. (ed.) Advances in Cryptology - CRYPTO 1991. LNCS, vol. 576, pp. 129–140. Springer, Heidelberg (1992)
27. Camenisch, J., Stadler, M.: Efficient group signature schemes for large groups. In: Kaliski Jr., B.S. (ed.) Advances in Cryptology - CRYPTO 1997. LNCS, vol. 1294, pp. 410–424. Springer, Heidelberg (1997)
28. Camenisch, J., Chaabouni, R., Shelat, A.: Efficient protocols for set membership and range proofs. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 234–252. Springer, Heidelberg (2008)
29. Camenisch, J., Kohlweiss, M., Rial, A., Sheedy, C.: Blind and anonymous identity-based encryption and authorised private searches on public key encrypted data. In: Jarecki, S., Tsudik, G. (eds.) PKC 2009. LNCS, vol. 5443, pp. 196–214. Springer, Heidelberg (2009)
30. Green, M., Hohenberger, S.: Blind identity-based encryption and simulatable oblivious transfer. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 265–282. Springer, Heidelberg (2007)
31. Han, J., Susilo, W., Mu, Y., Zhou, J., Au, M.H.: PPDCP-ABE: Privacy-Preserving Decentralized Cipher-Policy Attribute-Based Encryption, Cryptology ePrint Archive: Report 2014/470, `http://eprint.iacr.org/2014/470`