Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption

Jinguang Han, *Student Member*, *IEEE*, Willy Susilo, *Senior Member*, *IEEE*, Yi Mu, *Senior Member*, *IEEE*, and Jun Yan

Abstract—Decentralized attribute-based encryption (ABE) is a variant of a multiauthority ABE scheme where each authority can issue secret keys to the user independently without any cooperation and a central authority. This is in contrast to the previous constructions, where multiple authorities must be online and setup the system interactively, which is impractical. Hence, it is clear that a decentralized ABE scheme eliminates the heavy communication cost and the need for collaborative computation in the setup stage. Furthermore, every authority can join or leave the system freely without the necessity of reinitializing the system. In contemporary multiauthority ABE schemes, a user's secret keys from different authorities must be tied to his global identifier (GID) to resist the collusion attack. However, this will compromise the user's privacy. Multiple authorities can collaborate to trace the user by his GID, collect his attributes, then impersonate him. Therefore, constructing a decentralized ABE scheme *with* privacy-preserving remains a challenging research problem. In this paper, we propose a privacy-preserving decentralized key-policy ABE scheme where each authority can issue secret keys to a user independently without knowing anything about his GID. Therefore, even if multiple authorities are corrupted, they cannot collect the user's attributes by tracing his GID. Notably, our scheme only requires standard complexity assumptions (e.g., decisional bilinear Diffie-Hellman) and does not require any cooperation between the multiple authorities, in contrast to the previous comparable scheme that requires nonstandard complexity assumptions (e.g., q-decisional Diffie-Hellman inversion) and interactions among multiple authorities. To the best of our knowledge, it is *the first* decentralized ABE scheme with privacy-preserving based on standard complexity assumptions.

Index Terms—Attribute-based encryption, multiauthority, privacy-preserving extract protocol, access control, privacy

1 Introduction

In traditional access control schemes [1], [2], a central authority (CA) can control a user's access to sensitive data. We observed the following drawbacks in these schemes, especially in distributed systems. First, since a user's identity needs to be validated by the authority, in a large distributed system, it is a difficult task to manage numerous users identities. Second, all users must trust the central authority. If the authority is malicious, he can impersonate any user without being detected. Being different from the traditional access control schemes, attribute-based access control [3], [4] are the schemes that allow users to be validated by the descriptive attributes instead of their unique identities. Furthermore, a user can share his data by specifying an access structure so that all the users whose attributes satisfy it can access the data without knowing their identities. Therefore, attribute-based access control schemes are efficient primitives to share data with multiple users without knowing their identities. In order to reduce the trust on the central authority, some decentralized and distributed access control schemes are proposed [5], [6], [7], [8], [9]. Although, decentralized attribute-based access control schemes demonstrated lots of metrics, they seldom consider the user's privacy. Therefore, a user's attributes could be exposed to the malicious authorities. Thereafter, to provide a sound solution for sharing sensitive data with multiple users in distributed systems, a decentralized attribute-based access control with privacy-preserving scheme should be addressed.

In an open communication environment, such as the Internet, sensitive data must be encrypted prior to being transmitted. To achieve this, encryption schemes can be employed to protect the confidentiality of the sensitive data. Nevertheless, traditional encryption schemes cannot express a complex access policy, and additionally, the sender must know all the public keys of the receivers. Attribute-based encryption (ABE) introduced by Sahai and Waters [4] is a more efficient encryption scheme and it can express a complex access structure. In an ABE scheme, both the user's secret keys and the ciphertext are labeled with sets of attributes. The encrypter can encrypt a message under a set of attributes. Prior to decrypting the ciphertext, the receiver must obtain the secret (attribute) keys from the central authority. The receiver can decrypt the ciphertext and obtain the data if and only if there is a match between his secret keys and the attributes listed in the ciphertext. The original idea of ABE is to construct a fuzzy (error-tolerant) identity-based encryption (IBE) scheme [10], [11], [12], [13], [14].

- *J. Han is with the Centre for Computer and Information Security Research, School of Computer Science and Software Engineering, University of Wollongong, Northfields Avenue, Wollongong NSW2522, Australia and College of Science, Hohai University, Nanjing 210098, China. E-mail: jh843@uowmail.edu.au.*
- *W. Susilo and Y. Mu are with the Centre for Computer and Information Security Research, School of Computer Science and Software Engineering, University of Wollongong, Northfields Avenue, Wollongong NSW2522, Australia. E-mail: {wsusilo, ymu}@uow.edu.au.*
- *J. Yan is with the School of Information Systems and Technology, University of Wollongong, Northfields Avenue, Wollongong NSW2522, Australia. E-mail: jyan@uow.edu.au.*

Manuscript received 17 Oct. 2011; revised 5 Jan. 2012; accepted 17 Jan. 2012; published online 27 Jan. 2012.
Recommended for acceptance by A. Nayak.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2011-10-0779.
Digital Object Identifier no. 10.1109/TPDS.2012.50.

Published by the IEEE Computer Society

Since its seminal introduction, ABE as a special primitive has attracted a lot of attention in the research community. Essentially, there are two kinds of ABE schemes:

Key-Policy ABE (KP-ABE). In these schemes, the secret keys are associated with an access structure, while the ciphertext is labeled with a set of attributes [4], [5], [15], [16], [17].

Ciphertext-Policy ABE (CP-ABE). In these schemes, the ciphertext is associated with an access structure, while the secret keys are labeled with a set of attributes [3], [18], [19], [20], [21].

An access structure is employed to control users from accessing the protected resource in systems where users need to cooperate with multiple parties. A monotonic access structure is an access structure where, given a universal set $\mathbb{P}$, if a subset S' of $\mathbb{P}$ satisfies the access structure, all subsets S of $\mathbb{P}$ which contain S' satisfy the access structure. A (k,n)-threshold access structure is an access structure where, given a universal set $\mathbb{P}$ with $|\mathbb{P}| = n$, a subset S of $\mathbb{P}$ satisfies the access structure if and only if it contains at least k elements in $\mathbb{P}$. In an ABE scheme, an access structure is selected by the authority (in KP-ABE) or the encrypter (in CP-ABE) to control who can decrypt the ciphertext. For example, in KP-ABE, the authority specifies a (k,n)-threshold access structure and issues secret keys to users according to this access structure. An encrypter can encrypt a message under k-out-of-n attributes and lists them in the ciphertext. If a user holds a set of attributes which contains those listed in the ciphertext, he can use his secret keys to decrypt the ciphertext and obtain the message. However, if a user does not hold the required attributes specified in the ciphertext, he cannot decrypt the ciphertext.

The limitation of the original ABE scheme is that it can only express a threshold access structure. Goyal et al. proposed an ABE scheme [16] for fine-grained access policy where any monotonic access structure can be expressed by an access tree. In an access tree, there is a tree access structure where interior nodes consist of AND and OR gates and the leaves consist of the attributes. Each interior node x of the tree specifies a threshold gate (k_x, n_x), where n_x is the number of the children of x and $k_x \leq n_x$. Thereafter, when $k_x = n_x$, the gate is an AND gate. When $k_x = 1$, the gate is an OR gate. If a set of attributes satisfies the tree access structure, the corresponding secret keys can be used to reconstruct the secret embedded in the vertex of the tree. Subsequently, Ostrovsky et al. proposed an ABE scheme [17] with a nonmonotonic access structure where the secret keys are labeled with a set of attributes including not only the positive but also the negative attributes. Comparatively, ABE scheme with nonmonotonic access structure can express a more complicated access policy.

The first CP-ABE scheme was proposed by Bethencourt et al. [3], and it was proven to be secure in the generic group model. In contrast with KP-ABE, the access structure in CP-ABE is determined by the encrypter, instead of the CA. Therefore, the encrypter can decide who will decrypt the ciphertext; while, this is decided by the CA in the KP-ABE schemes. Cheung and Newport proposed another CP-ABE scheme [18] and reduced the difficulty of breaking their scheme to the decisional bilinear Diffie-hellman (DBDH) assumption. Both these CP-ABE schemes can only express a threshold access structure. Waters proposed a more generic CP-ABE scheme [21] where any access structure can be expressed by using the linear secret sharing scheme (LSSS) [22].

Attrapadung and Imai proposed a dual-policy ABE scheme [23] which combines a KP-ABE scheme with CP-ABE scheme. In this scheme, two access structures are created. One is for the objective attributes labeled with the ciphertext, and the other is for the subjective attributes held by the users. Furthermore, there is only one access structure in both KP-ABE and CP-ABE schemes.

Rial and Bart Preneel [24] proposed a blind key extract protocol for the centralized ABE scheme [3]. Hence, this scheme constitutes a blind centralized ABE scheme.

An ABE scheme should be secure against the collusion attacks [4], namely no group of users can combine their secret keys to decrypt the ciphertext which none of them can decrypt by himself. The most common technique to prevent collusion attacks is randomization. The central authority randomizes the user's secret keys by selecting random numbers [17], [18] or random polynomials [4], [5], [15].

ABE has been used as a building block to express flexible access structures in practical systems, such as distributed systems [25], data outsourcing systems [26], and cloud computing [27].

1.1 Multiple-Authority Attribute-Based Encryption

In their seminal work, Sahai and Waters left an open question that whether it is possible to construct an ABE scheme where the secret keys can come from multiple authorities [4]. Chase answered this question affirmatively by proposing a multiauthority KP-ABE scheme [5]. In this scheme, there are multiple authorities, one of which is called central authority. The central authority knows the secret keys of the other authorities. The user needs to obtain secret keys from all these authorities. Being different from one authority ABE, it is hard to resist collusion attacks in multiauthority ABE schemes. If the multiple authorities can work independently, the scheme is subject to this attack. Chase [5] overcame this problem by introducing the global identifier (GID) to the multiauthority ABE scheme. All the user's secret keys from different authorities must be tied to his GID. In order to let the ciphertext be independent of the user's GID, the central authority must compute a special secret key for the user using his secret key and the other authorities' secret keys. Although this scheme is *not* a decentralized ABE scheme, Chase made an important step from one authority ABE to multiauthority ABE.

Lin et al. proposed a multiauthority ABE scheme without a central authority [7] based on the distributed key generation (DKG) protocol [28] and the joint zero secret sharing (JZSS) protocol [29]. 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.

Chase and Chow proposed another multiauthority KP-ABE scheme [15] which improved the previous scheme [5] and removed the need of a central authority. Notably, they also addressed the privacy of the user. In previous multi-authority ABE schemes [5], [7], the user must submit his GID to each authority to obtain the corresponding secret keys. This will risk the user being traced by a group of corrupted authorities. Chase and Chow provided an anonymous key issuing protocol for the GID where a 2-party secure computation technique is employed. As a result, a group of authorities cannot cooperate to pool the user's attributes by tracing his GID. However, the multiple authorities must collaborate to setup the system. Each pair of authorities must execute a 2-party key exchange protocol to share the seeds of the selected pseudorandom functions (PRF) [30]. 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. The security of this scheme can be reduced to DBDH assumption and no-standard complexity assumption (q-decisional Diffie-Hellman inverse (q-DDHI)). Chase and Chow also left an open challenging research problem on how to construct a privacy-preserving multiauthority ABE scheme without the need of cooperations among the authorities.

Lekwo and Waters proposed a new multiauthority ABE scheme named decentralizing CP-ABE scheme [8]. This scheme improved the previous multiauthority 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. Note that the 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. As mentioned in [21], unfortunately this scheme is inefficient. Furthermore, the attributes of the user can be collected by tracing his GID.

Müller et al. proposed a distributed CP-ABE scheme [6], [31], where the pairing operations executed in the decryption stage are constant. This scheme was proven to be secure in the generic group [3], instead of reducing to a complexity assumption. Furthermore, there must be a central authority who generates the global key and issues secret keys to the user.

Liu et al. proposed a fully secure multiauthority CP-ABE scheme [32] in the standard model. Their scheme is based on the CP-ABE scheme [20]. In their scheme, there are multiple central authorities and attribute authorities. The central authorities issue identity-related keys to users and the attribute authorities issue attribute-related keys to users. Prior to obtaining attribute keys from the attribute authorities, the user must obtain secret keys from multiple central authorities. This multiauthority ABE scheme was also designed in the composite order $(N = p_1p_2p_3)$ bilinear group.

Li et al. [9] proposed a multiauthority cipher-policy ABE scheme with accountability, where the anonymous key issuing protocol [15] was employed. In this scheme, the user can *only* obtain secret keys anonymously from $N-1$ authorities; while he can be traced when he shared his secret keys with others. Unfortunately, the multiple authorities must initialize the system interactively. Their scheme relied on DBDH assumption, decisional linear (DLIN) assumption and q-DDHI assumption.

1.2 Our Contribution

In this paper, we answered the question left by Chase and Chow [15] affirmatively by proposing a decentralized KP-ABE scheme with the privacy-preserving key extraction protocol. In our scheme, multiple authorities can work independently without any cooperation and a central authority. The GID is used to tie all the user's secret keys together, while the corrupted authorities cannot pool the user's attributes by tracing it. Our scheme is *any* number resilient for the users and $(N-1)$-tolerant for the authorities, where N is the number of the authorities in the system. Our privacy-preserving decentralized ABE scheme is based on standard complexity assumption (DBDH), instead of any nonstandard complexity assumptions (e.g., q-decisional Diffie-Hellman inversion). To the best of our knowledge, it is *the first* decentralized ABE scheme with privacy-preserving that is based on merely a standard assumption.

1.3 Paper Organization

In Section 2, we review the preliminaries used throughout this paper. Subsequently, a privacy-preserving decentralized ABE scheme is proposed and proven in Section 3. Finally, Section 4 concludes the paper.

2 Preliminaries

In the rest of this paper, by $x \overset{R}{\leftarrow} X$, we denote that x is randomly selected from X. Especially, by $x \overset{R}{\leftarrow} X$, we denote that x is selected from X identically if X is a finite set. We say that a function $\epsilon : \mathbb{Z} \rightarrow \mathbb{R}$ is negligible, if for all $z \in \mathbb{Z}$ there exists a value $\eta \in \mathbb{Z}$ such that $\epsilon(x) < \frac{1}{x^z}$ for all $x > \eta$. By $R \overset{s}{\leftarrow} S$ and $R \overset{r}{\rightarrow} S$, we denote that party S sends s to party R and party R sends r to party S, respectively. We denote $\mathcal{KG}(1^\ell)$ as the secret-public key generation algorithm where ℓ is the security parameter. If X is a finite set, by $|X|$, we denote the cardinality of X. By $A(x) \rightarrow y$, we denote that y is computed by running algorithm A on input x. Suppose that $\mathbb{Z}_p$ is a finite field with prime order p, by $\mathbb{Z}_p[x]$, we denote the polynomial ring on $\mathbb{Z}_p$, which consists of all polynomials with coefficients from $\mathbb{Z}_p$.

2.1 Building Blocks

In this paper, the following building blocks are used.

Lagrange Interpolation. Suppose that $p(x) \in \mathbb{Z}_p[x]$ is a $(k-1)$-degree polynomial. Given k different polynomial values $p(x_1), p(x_2), \ldots, p(x_k)$, the polynomial $p(x)$ can be reconstructed as follows:

$$p(x) = \sum_{x_i \in S} p(x_i) \prod_{x_j \in S, x_j \neq x_i} \frac{x - x_j}{x_i - x_j} = \sum_{x_i \in S} p(x_i) \Delta_{x_i,S}(x),$$

where $S = \{x_i, x_2, \ldots, x_k\}$. The Lagrange coefficient for x_i in S is

$$\Delta_{x_i,S}(x) = \prod_{x_j \in S, x_j \neq x_i} \frac{x - x_j}{x_i - x_j}.$$

Therefore, given any k different values $p(x_1)$, $p(x_2), \ldots,$ $p(x_k)$, we can compute $p(x)$ for $\forall x \in \mathbb{Z}_p$. However, when only $k-1$ different polynomial values are provided, the other polynomial values are unconditionally hidden.

Commitment. A commitment scheme consists of tree algorithms: $\mathcal{C} = (\mathrm{Setup}, \mathrm{Commit}, \mathrm{Decommit})$.

- Setup$(1^\ell) \rightarrow params$. This algorithm takes as input a security parameters ℓ and outputs the system parameters $params$.
- Commit$(params, M) \rightarrow (com, decom)$. This algorithm takes as input the parameters $params$ and a message M and outputs a commitment com and a decommitment $decom$. $decom$ can be used to decommite the commitment com.
- Decommit$(params, M, com, decom) \rightarrow \{0, 1\}$. This algorithm takes as input the parameter $params$, the message M, the commitment com and the decommitment $decom$ and outputs 1 if $decom$ can decommite com to M; Otherwise, this algorithm outputs 0.

A commitment scheme should satisfy two properties: hiding and binding. The hiding property requires that the message M keeps undisclosed until the user reveals it. The binding property requires that only one value $decom$ can be used to decommit the commitment.

We use the Pedersen commitment scheme [33] which is a perfectly hiding commitment scheme and is based on the discrete logarithm assumption. Let G be a prime order group with generators $g_0, g_1, g_2, \ldots, g_l$. In order to commit messages $(m_1, m_2, \ldots, m_l)$, the user selects $r \overset{R}{\leftarrow} \mathbb{Z}_p$, and computes the commitment $T = g_0^r \prod_{j=1}^{l} g_j^{m_j}$. The user can use r to decommit the commitment later.

Proof of Knowledge. We use the notation introduced by Camenisch and Stadler [34] to prove statements about discrete logarithm. By

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

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 simultaneously in groups $G = \langle g \rangle = \langle h \rangle$ and $\tilde{G} = \langle \tilde{g} \rangle = \langle \tilde{h} \rangle$. Conventionally, the values in the parenthesis denote the knowledge that is being proven, while the rest of the other values are known to the verifier. There exists a knowledge extractor which can be used to rewind these quantities from a successful prover.

2.2 Decentralized Key-Policy Attribute-Based Encryption

The formal definition of access structure is as follows:

Definition 1 (Access Control) [22]. *Let $\mathbb{P} = \{P_1, P_2, \ldots, P_N\}$ be a set of parties. A collection $\mathbb{A} \subseteq 2^{\{P_1, P_2, \ldots, P_N\}}$ is monotonic, if $S_1 \in \mathbb{A}$ and $S_1 \subseteq S_2$ implies $S_2 \in \mathbb{A}$. An access structure (resp., monotonic access structure) is a collection (resp., monotonic collection) $\mathbb{A}$ of nonempty subsets of $\{P_1, P_2, \ldots, P_N\}$, namely $\mathbb{A} \subseteq 2^{\{P_1, P_2, \ldots, P_N\}} \setminus \{\phi\}$. The sets in $\mathbb{A}$ are called the authorized sets, and the sets outside of $\mathbb{A}$ are called the unauthorized sets.*

A decentralized KP-ABE scheme consists of the following five algorithms:

- Global Setup$(1^\ell) \rightarrow params$. This algorithm takes as input a security parameter ℓ and outputs the system parameters $params$.
- Authority Setup$(1^\ell) \rightarrow (SK_i, PK_i, \mathbb{A}_i)$. Each authority A_i generates his secret-public key pair $\mathcal{KG}(1)^\ell \rightarrow (SK_i, PK_i)$ and an access structure $\mathbb{A}_i$, for $i = 1, 2, \ldots, N$.
- KeyGen$(SK_i, GID, A^i_{GID}) \rightarrow SK^i_U$. Each authority A_i takes as input his secret key SK_i, a global identifier GID and a set of attributes A^i_{GID}, and outputs the secret keys SK^i_U, where $A^i_{GID} = A_{GID} \bigcap \tilde{A}_i$, A_{GID} and $\tilde{A}_i$ denote the attributes corresponding to the GID and monitored by A_i, respectively.
- Encryption$(params, M, A_C) \rightarrow CT$. This algorithm takes as input the system parameters $params$, a message M and a set of attributes A_C, and outputs the ciphertext CT, where $A_C = \{A^1_C, A^2_C, \ldots, A^N_C\}$ and $A^i_C = A_C \bigcap \tilde{A}_i$.
- Decryption$(GID, \{SK^i_U\}_{i \in I_C}, CT)$. This algorithm takes as input the global identifier GID, the secret keys $\{SK^i_U\}_{i \in I_C}$ and the ciphertext CT, and outputs the message M, where I_C is the index set of the authorities A_i such that $A^i_C \neq \{\phi\}$.

Definition 2. *We say that a decentralized key-policy attribute-based encryption scheme is correct if*

$$\Pr\left[\begin{array}{l|l} & \text{Global Setup}(1^\ell) \\ & \rightarrow params; \\ & \text{Authorities Setup}(1^\ell) \\ \text{Decryption}(GID, & \rightarrow (SK_i, PK_i, \mathbb{A}_i); \\ \{SK^i_U\}_{i \in I_C}, CT) & \text{KeyGen}(SK_i, GID, A^i_{GID}) \\ = M & \rightarrow SK^i_U; \\ & \text{Encryption(params, M, A}_\text{C}) \\ & \rightarrow CT; \\ & \{A_{GID} \bigcap \tilde{A}_i \in \mathbb{A}_i\}_{i \in I_C}, \end{array}\right] = 1$$

where the probability is taken over the random coins of all the algorithms in the protocol.

Security Model

Our security model on the decentralized ABE is similar to the model proposed in [5], [15], which is known as the selective-set model. This model is described as follows:

Initialization. The adversary $\mathcal{A}$ submits a set of attributes A_C which he wants to be challenged and a list of corrupted authorities $C_\mathcal{A}$, where $|C_\mathcal{A}| < N$. There should exist at least one authority A_j such that $A_C \bigcap \tilde{A}_j \notin \mathbb{A}_j$.

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

Authority setup.

1. For $A_i \in C_\mathcal{A}$, the challenger sends the secret-public key pair (SK_i, PK_i) to $\mathcal{A}$.
2. For $A_i \notin C_\mathcal{A}$, the challenger sends the public key PK_i to $\mathcal{A}$.

Phase 1. The adversary $\mathcal{A}$ can query secret keys for sets of attributes $A^*_{GID_1}, A^*_{GID_2}, \ldots, A^*_{GID_{q_1}}$, the only constraint is $A_C \not\subseteq A^*_{GID_i}$ for $i = 1, 2, \ldots, q_1$.

Challenge. $\mathcal{A}$ submits two messages M_0 and M_1 with equal length. The challenger flips an unbiased coin with $\{0, 1\}$, and obtain $b \in \{0, 1\}$. The challenger computes $CT^* = \text{Encryption}(params, M_b, A_C)$ and sends CT^* to $\mathcal{A}$.

Phase 2. The adversary $\mathcal{A}$ can query secret keys for sets of attributes $A^*_{GID_{q_1+1}}, A^*_{GID_{q_1+2}}, \ldots, A^*_{GID_q}$. Phase 1 is repeated.

Guess. The adversary $\mathcal{A}$ outputs his guess b' on b.

Definition 3. *A decentralized key-policy attribute-based encryption (DKP-ABE) scheme is (T, q, ϵ) secure in the selective-set model if no probabilistic polynomial-time adversary $\mathcal{A}$ making q secret key queries has advantage at least*

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

in the selective-set model.

2.3 Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption

We have described the decentralized KP-ABE scheme and its security model. The privacy-preserving decentralized KP-ABE scheme has the same algorithms Global Setup, Authority Setup, Encryption, and Decryption with the decentralized KP-ABE scheme. We only replace the algorithm KeyGen in the decentralized KP-ABE scheme with algorithm BlindKeyGen. In a privacy-preserving decentralized KP-ABE scheme, the authorities do not know the user's GID nor can cause failures using the information of the GID. This concept is from blind IBE schemes [35], [36]. We define this algorithm as follows:

BlindKeyGen

$$(U(params,\ PK_i,\ GID,\ decom) \leftrightarrow A_i(params, SK_i, PK_i, \mathbb{A}_i, com)) \rightarrow \left(SK_U^i, \text{empty}\right).$$

In this algorithm, the user U runs the commitment algorithm $\text{Commit}(params, GID) \rightarrow (com, decom)$ and sends com to the authority A_i. Then, the user U and the authority A_i take as input $(params, PK_i, GID, decom)$ and

$$(params, SK_i, PK_i, \mathbb{A}_i, com),$$

respectively. If $\text{Decommit}(params, GID, com, decom) \rightarrow 1$, this algorithm outputs a secret key SK_U^i for U and empty for A_i. Otherwise it outputs error messages $(\perp, \perp)$ for both U and A_i.

The algorithm BlindKeyGen should satisfy the following two properties: leak-freeness and selective-failure blindness [35], [36]. Leak-freeness requires that, by executing algorithm BlindKeyGen with the honest authorities, the malicious user cannot know anything which he cannot learn by executing algorithm KeyGen with the honest authorities. Selective-failure blindness requires that the malicious authorities cannot know anything about the user's GID and cannot cause the algorithm BlindKeyGen to selectively fail depending on the user's choice of GID. We use the following two games to define these two properties.

Leak-freeness. This game is defined by the real experiment and the ideal experiment:

Real Experiment. Runs $\text{Setup}(1^\ell) \rightarrow params$ and Authority Setup$(1^\ell) \rightarrow (SK_i, PK_i, \mathbb{A}_i)$. As many times as the distinguisher $\mathcal{D}$ wants, the adversary $\mathcal{U}$ chooses a GID and executes the algorithm BlindKeyGen with the authority

$$A_i : \text{BlindKeyGen}(U(params, PK_i, GID, decom) \leftrightarrow A_i(params, SK_i, PK_i, \mathbb{A}_i, com)).$$

Ideal Experiment. Runs $\text{Setup}(1^\ell) \rightarrow params$ and Authority Setup$(1^\ell) \rightarrow (SK_i, PK_i, \mathbb{A}_i)$. As many times as the distinguisher $\mathcal{D}$ wants, the simulator $\hat{\mathcal{U}}$ chooses a GID and queries a trusted party to obtain the output of the algorithm KeyGen(SK_i, GID, A_{GID}^i) if $\text{Decommit}(params, GID, com, decom) \rightarrow 1$, and $\perp$ otherwise.

Definition 4. *An algorithm $\text{BlindKeyGen}(U \leftrightarrow A_i)$ associated with a decentralized KP-ABE scheme $\prod =$ (Global Setup, Authority Setup, KeyGen, Encryption, Decryption) is leak-free if for all efficient adversaries $\mathcal{U}$, there exists an efficient simulator $\hat{\mathcal{U}}$ such that for the security parameter ℓ, no efficient distinguisher $\mathcal{D}$ can distinguish whether $\mathcal{U}$ is executing Real Experiment or Ideal Experiment with nonnegligible advantage.*

Selective-failure Blindness. This game is defined as follows:

1. The adversary authority $\mathcal{A}_i$ outputs his public key PK_i and a pair of global identifiers (GID_0, GID_1).
2. A random bit $b \in \{0, 1\}$ is selected randomly.
3. $\mathcal{A}_i$ is given two commitments
$$com_b = \text{Commit}(params, GID_b)$$
and $com_{1-b} = \text{Commit}(params, GID_{1-b})$, and black-box access the two oracles $U(params, PK_i, GID_b, com_b)$ and $U(params, PK_i, GID_{1-b}, com_{1-b})$.
4. The algorithm U outputs $SK_{U,b}^i$ and $SK_{U,1-b}^i$, respectively.
5. If $SK_{U,b}^i \neq \perp$ and $SK_{U,1-b}^i \neq \perp$, $\mathcal{A}_i$ is given
$$(SK_{U,b}^i, SK_{U,1-b}^i).$$
If $SK_{U,b}^i \neq \perp$ and $SK_{U,1-b}^i = \perp$, $\mathcal{A}_i$ is given $(\epsilon, \perp)$. If $SK_{U,b}^i = \perp$ and $SK_{U,1-b}^i \neq \perp$, $\mathcal{A}_i$ is given $(\perp, \epsilon)$. If $SK_{U,b}^i = \perp$ and $SK_{U,1-b}^i = \perp$, $\mathcal{A}_i$ is given $(\perp, \perp)$.
6. Finally, $\mathcal{A}_i$ outputs his guess b' on b.

Definition 5. *An algorithm $\text{BlindKeyGen}(U \leftrightarrow A_i)$ associated with a decentralized KP-ABE scheme $\prod =$ (Global Setup, Authority Setup, KeyGen, Encryption, Decryption) is selective-failure blind if no probabilistic polynomial-time adversary has advantage $Avd_{\mathcal{A}_i}^{SFB} = |Pr[b' = b] - \frac{1}{2}| > \epsilon(\ell)$ in the above game.*

Definition 6 (Privacy-Preserving Decentralized KP-ABE). *A privacy-preserving decentralized KP-ABE scheme $\tilde{\prod} =$ (Global Setup, Authority Setup, BlindKeyGen, Encryption, Decryption) is secure in the selective-set model if and only if: 1)*

$$\prod = (\text{Global Setup, Authority Setup, KeyGen, Encryption, Decryption})$$

is a secure decentralized KP-ABE scheme in the selective-set model; and 2) algorithm BlindKeyGen is leak-free and selective-failure blind.

2.4 Complexity Assumption

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

1. Bilinearity. for all $x, y \in \mathbb{G}$ and $u, v \in \mathbb{Z}_p$, $e(x^u, y^v) = e(x, y)^{uv}$.
2. Nondegeneracy. $e(g, g) \neq 1$, where 1 is the identity of $\mathbb{G}_\tau$.

- **Global Setup.** This algorithm takes as input the security parameter ℓ, and outputs a bilinear group $(e, p, \mathbb{G}, \mathbb{G}_\tau) \leftarrow \mathcal{GG}(1^\ell)$ with prime order p, where $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_\tau$. Let g, h, and h_1 be the generators of $\mathbb{G}$. Suppose that there are N authorities $A_1, A_2, \cdots, A_N$ in the system. A_i monitors a set of attributes $\tilde{A}_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,n_i}\}$, for $i = 1, 2, \cdots, N$. Let the set of universal attributes $\mathcal{U} = \bigcup_{i=1}^N \tilde{A}_i$.
- **Authorities Setup.** Each authority A_i generates his secret-public key pair $(\alpha_i, \beta_i, Y_i, Z_i) \leftarrow \mathcal{KG}(1^\ell)$, where $Y_i = e(g, g)^{\alpha_i}$ and $Z_i = g^{\beta_i}$. For each $a_{i,j} \in \tilde{A}_i$, it chooses $t_{i,j} \xleftarrow{R} \mathbb{Z}_p$, and computes $T_{i,j} = g^{t_{i,j}}$. The public keys and secret keys of A_i are $PK_i = \{Y_i, Z_i, T_{i,1}, T_{i,2}, \cdots, T_{i,n_i}\}$ and $SK_i = \{\alpha_i, \beta_i, t_{i,1}, t_{i,2}, \cdots, t_{i,n_i}\}$, for $i = 1, 2, \cdots, N$. Each authority A_i specifies an (k_i, n_i)-threshold access structure $\mathbb{A}_i$, where $k_i \leq n_i$.
- **KeyGen.** Suppose that a user U has the global identifier $u \in \mathbb{Z}_p$ and a set of attributes A_U. To generate a key for U for the attribute $a_{i,j} \in \tilde{A}_i$, A_i chooses $r_i \xleftarrow{R} \mathbb{Z}_p$, and a $(k_i - 1)$-degree polynomial $p_i(x) \xleftarrow{R} \mathbb{Z}_p[x]$ with $p_i(0) = r_i$, and computes
$$D_i = g^{\alpha_i} h^{r_i} h_1^{u\beta_i},\ D_{i,j} = h^{\frac{p_i(a_{i,j})}{t_{i,j}}},\ \text{for } a_{i,j} \in A_U^i$$
where $A_U^i = A_U \bigcap \tilde{A}_i$, for $i = 1, 2, \cdots, N$.
- **Encryption.** This algorithm takes as input a set of attributes $A_C = \{A_C^1, A_C^2, \cdots, A_C^N\}$ and a random number $s \xleftarrow{R} \mathbb{Z}_p$, and outputs the ciphertext as follows
$$C_1 = M \cdot \prod_{i \in I_C} e(g, g)^{\alpha_i s},\ C_2 = g^s,\ C_3 = \prod_{i \in I_C} g^{\beta_i s},\ \{C_{i,j} = T_{i,j}^s\}_{a_{i,j} \in A_C}$$
where $A_C^i = A_C \bigcap \tilde{A}_i$ and I_C is the index set of the authorities A_i such that $A_C^i \neq \{\phi\}$, for $i = 1, 2, \cdots, N$.
- **Decryption.** To decrypt the ciphertext $C = (C_1,\ C_2,\ C_3,\ \{C_{i,j}\}_{a_{i,j} \in A_C})$, the user computes $E = \prod_{i \in I_C} e(D_i, C_2),\ V = e(C_3, h_1^u),\ F_i = \prod_{a_{i,j} \in A_C^i} e(C_{i,j}, D_{i,j})^{\Delta_{a_{i,j}, A_C^i}(0)} (i \in I_C)$ and $M = C_1 \cdot \frac{V \cdot \prod_{i \in I_C} F_i}{E}$

Fig. 1. Decentralized key-policy attribute-based encryption.

3. **Computability.** There exists an efficient algorithm to compute $e(x, y)$ for all $x, y \in \mathbb{G}$.

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

Definition 7 (Decisional Bilinear Diffie-Hellman Assumption) [10]. *Let $a, b, c, z \xleftarrow{R} \mathbb{Z}_p$, $\mathcal{GG}(1^\ell) \rightarrow (e, p, \mathbb{G}, \mathbb{G}_\tau)$, and g be a generator of $\mathbb{G}$. The DBDH assumption holds in $(e, p, \mathbb{G}, \mathbb{G}_\tau)$, if no probabilistic polynomial-time adversary $\mathcal{A}$ can distinguish $(A, B, C, Z) = (g^a, g^b, g^c,\ e(g, g)^{abc})$ from $(A, B, C, Z) = (g^a, g^b, g^c, e(g, g)^z)$ with advantage*
$$Adv_{\mathcal{A}}^{DBDH} = \left| \begin{matrix} Pr[\mathcal{A}(A, B, C, e(g, g)^{abc}) = 1] \\ -Pr[\mathcal{A}(A, B, C, e(g, g)^z) = 1] \end{matrix} \right| > \epsilon(\ell),$$
where the probability is taken over the random choice of $a, b, c, z \xleftarrow{R} \mathbb{Z}_p$, and the bits consumed by $\mathcal{A}$.

3 Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption

In this section, we propose a decentralized KP-ABE scheme based on the DBDH assumption. Then, we describe a privacy-preserving extract protocol for the secret keys.

In our privacy-preserving decentralized KP-ABE scheme, a user executes a 2-party secure computation protocol with an authority to obtain his secret keys. As a result, the user can obtain his secret keys anonymously without releasing anything about his identifier to the multiple authorities. As pointed in [15], an anonymous credential system [37], [38] can be used by the user to convince the authorities that he holds the corresponding attributes without revealing his identifier. In an anonymous credential system, a user can obtain a credential and prove the possession anonymously. The user can interact with different partners with different pseudonyms [39] such that no partner can link the pseudonyms to the same user. Furthermore, the user can prove that he has obtained multiple credentials which correspond to the same identifier without revealing it. Hence, this technique can be employed in our system to allow the user to obtain the corresponding secret keys without revealing his identifier to the authorities.

3.1 Decentralized Key-Policy Attribute-Based Encryption

Our decentralized KP-ABE scheme is described in Fig. 1. This idea is inspired by the IBE schemes [13], [14] and the multiauthority ABE schemes [5], [8], [15].

Overview. In our scheme, suppose that there are N authorities $A_1, A_2, \ldots, A_N$. Authority A_i manages a set of attributes $\tilde{A}_i = \{a_{i,1}, a_{i,2}, \ldots, a_{i,n_i}\}$ and specifies an (k_i, n_i)-threshold access structure $\mathbb{A}_i$, for $i = 1, 2, \ldots, N$. A_i generates a secret-public key pair $((\alpha_i, \beta_i), (Y_i, Z_i))$ and publishes (Y_i, Z_i). For each attribute $a_{i_j} \in \tilde{A}_i$, A_i creates a secret-public key pair $(t_{i,j}, T_{i,j})$ and publishes $T_{i,j}$. The secret keys and public keys of A_i are $(\alpha_i, \beta_i, \{t_{i,j}\}_{a_{i,j} \in \tilde{A}_i})$ and $(Y_i, Z_i, \{T_{i,j}\}_{a_{i,j} \in \tilde{A}_i})$, respectively. To issue secret keys to a user U with a set of attributes A_U, the authority A_i selects a random number $r_i \xleftarrow{R} \mathbb{Z}_p$ and computes D_i using r_i, his secret key (α_i, β_i) and the user's identifier u. Hence, the user's identifer u is tied to his secret keys. We use D_i to protect against the collusion attacks. Otherwise, two users with identifier u_1 and u_2 can combine their secret keys from A_i and A_j together. The authority A_i selects a $(k_i - 1)$-degree polynomial $p_i(x)$ with $p_i(0) = r_i$. For each attribute $a_{i,j} \in A_U \bigcap \tilde{A}_i$, A_i computes the secret key $D_{i,j}$ using the value $p_i(a_{i,j})$ and $t_{i,j}$. To encrypt a message $M \in \mathbb{G}_\tau$ under a set of attributes $A_C = \{A_C^1, A_C^2, \dots, A_C^N\}$ where $A_C^i = A_C \bigcap \tilde{A}_i$ for $i = 1, 2, \dots, N$, a random number $s \xleftarrow{R} \mathbb{Z}_p$ is selected to hide M in $C_1 = M \cdot \prod_{i \in I_C} e(g,g)^{s\alpha_i}$. The ciphertext is $C = (C_1, C_2, C_3, \{C_{i,j}\}_{a_{i,j} \in A_C})$, where $\{C_{i,j}\}_{a_{i,j} \in A_C}$ are computed by using s and the public keys $\{T_{i,j}\}_{a_{i,j} \in A_C}$. If a user whose attributes contain those listed in the ciphertext, he can use his secret keys D_i and C_2 to compute

$$E = \prod_{i \in I_C} e(D_i, C_2) = \prod_{i \in I_C} e(g,g)^{s\alpha_i} \prod_{i \in I_C} e(g,h)^{sr_i} \prod_{i \in I_C} e(g,h_1)^{us\beta_i}$$

and use $\{D_{i,j}\}_{a_{i,j} \in A_C}$ and $\{C_{i,j}\}_{a_{i,j} \in A_C}$ to reconstruct the exponential r_i and compute $F_i = \prod_{i \in I_C} e(g,h)^{sr_i}$. Using C_3 and his identifier u, the user can compute

$$V = \prod_{i \in I_C} e(g,h_1)^{us\beta_i}.$$

So, the user can obtain $\prod_{i \in I_C} e(g,g)^{s\alpha_i}$ by removing $\prod_{i \in I_C} F_i$ and V from E. At the end, he can obtain M by removing $\prod_{i \in I_C} e(g,g)^{s\alpha_i}$ from C_1.

Correctness. We have

$$\begin{aligned} E &= \prod_{i \in I_C} e(D_i, C_2) \\ &= \prod_{i \in I_C} e(g^{\alpha_i} h^{r_i} h_1^{u\beta_i}, g^s) \\ &= \prod_{i \in I_C} e(g,g)^{s\alpha_i} \prod_{i \in I_C} e(g,h)^{sr_i} \prod_{i \in I_C} e(g,h_1)^{us\beta_i}, \\ V &= e(C_3, h_1^u) = \prod_{i \in I_C} e(g,h_1)^{us\beta_i}, \end{aligned}$$

and

$$\begin{aligned} F_i = & \prod_{a_{i,j} \in A_C^i} e(C_{i,j}, D_{i,j})^{\Delta_{a_{i,j}, A_C^i}(0)} \\ = & \prod_{a_{i,j} \in A_C^i} e(g^{st_{i,j}}, h^{\frac{p_i(a_{i,j})}{t_{i,j}}})^{\Delta_{a_{i,j}, A_C^i}(0)} \\ = & \prod_{a_{i,j} \in A_C^i} e(g,h)^{sp_i(a_{i,j})\Delta_{a_{i,j}, A_C^i}(0)} \\ & = e(g,h)^{s \sum_{a_{i,j} \in A_C^i} p_i(a_{i,j}) \Delta_{a_{i,j}, A_C^i}(0)} \\ & = e(g,h)^{sr_i}. \end{aligned}$$

Therefore,

$$\begin{aligned} & C_1 \cdot \frac{V \cdot \prod_{i \in I_C} F_i}{E} \\ & = M \cdot \frac{\prod_{i \in I_C} e(g,g)^{s\alpha_i} e(g,h_1)^{us\beta_i} e(g,h)^{sr_i}}{\prod_{i \in I_C} e(g,g)^{s\alpha_i} e(g,h)^{sr_i} e(g,h_1)^{us\beta_i}} \\ & = M. \end{aligned}$$

Theorem 1. *Our decentralized key-policy attribute-based encryption scheme is* $(\Gamma, q, \epsilon(\ell))$ *semantically secure (CPA) in the selective-set model, if the* $(\Gamma', \epsilon'(\ell))$ *decisional bilinear Diffie-Hellman assumption holds in* $(e, p, \mathbb{G}, \mathbb{G}_\tau)$*, where*

$$\Gamma' = \Gamma + \mathcal{O}(\Gamma) \textit{ and } \epsilon'(\ell) = \frac{1}{2}\epsilon(\ell).$$

Proof. Suppose that there exists an adversary $\mathcal{A}$ who can $(\Gamma, q, \epsilon(\ell))$ break our decentralized KP-ABE scheme, there will exist an algorithm $\mathcal{B}$ that can use $\mathcal{A}$ to break the decisional bilinear Diffie-Hellman assumption as follows:

The challenger generates the bilinear group $(e, p, \mathbb{G}, \mathbb{G}_\tau)$, and chooses a random generator $g \in \mathbb{G}$. He flips an unbiased coin μ with $\{0, 1\}$. If $\mu = 0$, he sends $(A, B, C, Z) = (g^a, g^b, g^c, e(g,g)^{abc})$ to $\mathcal{B}$. Otherwise, he sends $(A, B, C, Z) = (g^a, g^b, g^c, e(g,g)^z)$ to $\mathcal{B}$, where $z \xleftarrow{R} \mathbb{Z}_p$. $\mathcal{B}$ will output his guess μ' on μ.

Initialization. The adversary $\mathcal{A}$ submits a set of attributes $A_C = \{A_C^1, A_C^2, \dots, A_C^N\}$ which he wants to be challenged and a list of corrupted authorities $C_\mathcal{A}$. Suppose that A_C is mapped to the user with global identifier u^*. $\mathcal{B}$ chooses $\gamma, \eta \xleftarrow{R} \mathbb{Z}_p$, and sets $h = Ag^\gamma$ and $h_1 = g^\eta$.

Authorities Setup. There should be at least one authority $\mathbf{A} \notin C_\mathcal{A}$ where the adversary can only get secret keys less than the specified threshold value. Suppose that $\mathbf{A}$ specifies the threshold (k, n) and $|A_C \bigcap \tilde{\mathbf{A}}| = k - 1$.

1. For $A_k \in C_\mathcal{A}$, it chooses $v_k, \beta_k, w_{k,j} \xleftarrow{R} \mathbb{Z}_p$, and sets:

   $$Y_k = e(g,g)^{v_k}, Z_k = g^{\beta_k}, \text{ and } \{T_{k,j} = g^{w_{k,j}}\}_{a_{k,j} \in \tilde{A}_k}.$$

   This implies that the secret key for $A_k \in C_\mathcal{A}$ is $(v_k, \beta_k, w_{k,j})$. $\mathcal{B}$ sends $(v_k, \beta_k, w_{k,j})$ and $(Y_k, Z_k, T_{k,j})$ to $\mathcal{A}$.
2. For $A_k \notin C_\mathcal{A}$ and $A_k \neq \mathbf{A}$, it chooses

   $$v_k, \beta_k, w_{k,j} \xleftarrow{R} \mathbb{Z}_p,$$

   and sets

   $$\begin{aligned} & Y_k = e(g,g)^{bv_k}, Z_k = g^{\beta_k}, \\ & \{T_{k,j} = h^{w_{k,j}} = g^{(a+\gamma)w_{k,j}}\}_{a_{k,j} \in \tilde{A}_i - A_C}, \end{aligned}$$

   and

   $$\{T_{k,j} = g^{w_{k,j}}\}_{a_{k,j} \in A_C \bigcap \tilde{A}_i}.$$

   This implies that the secret key for $A_k \notin C_\mathcal{A}$ is $(bv_k, \beta_k, w_{k,j})$. $\mathcal{B}$ sends $(Y_k, Z_k, T_{k,j})$ to $\mathcal{A}$.
3. For $\mathbf{A}$, it chooses $w_j, \beta \xleftarrow{R} \mathbb{Z}_p$, and sets

   $$\begin{aligned} & Y = e(g,g)^{ab} \prod_{A_k \notin C_\mathcal{A}} e(g,g)^{-v_k b} \prod_{A_k \in C_\mathcal{A}} e(g,g)^{-v_k}, \\ & Z = g^\beta, \{T_j = g^{w_j}\}_{a_j \in A_C \bigcap \tilde{\mathbf{A}}}, \end{aligned}$$

   and

   $$\{T_j = h^{w_j} = g^{(a+\gamma)w_j}\}_{a_j \in \tilde{\mathbf{A}} - A_C}.$$

   This implies that the secret key for $\mathbf{A}$ is (v, β, ω_j), where $v = ab - \sum_{A_k \notin C_\mathcal{A}} v_k b - \sum_{A_k \in C_\mathcal{A}} v_k$. $\mathcal{B}$ sends (Y, Z, T_j) to $\mathcal{A}$.

TABLE 1
The Comparison of Computing Cost

Schemes	Authority setup	KeyGen	Encryption	Decryption
Chase's scheme [5]	$(\lvert\mathcal{U}\rvert+1)E$	$(\lvert A_U\rvert+1)E$	$(\lvert A_C\rvert+2)E$	$\lvert A_C\rvert E+(\lvert A_C\rvert+1)P$
MKE's scheme [6], [31]	$2\lvert\mathcal{U}\rvert E$	$\lvert A_U\rvert E$	$3\lvert I_C\rvert E$	$2P$
CC scheme [15]	$(\lvert\mathcal{U}\rvert+2N)E$	$(\lvert\mathcal{U}\rvert+\lvert I_U\rvert^2)E$	$(\lvert A_C\rvert+2)E$	$\lvert A_C\rvert E+(\lvert A_C\rvert+1)P$
LW's scheme [8]	$2NE$	$2\lvert A_U\rvert E$	$(5\lvert A_C\rvert+1)E$	$3\lvert A_C\rvert(E+P)$
LCHWY scheme [32]	$(\lvert\mathcal{U}+N\rvert E)$	$(4d+\lvert A_U\rvert)E+\lvert I_U\rvert P$	$(3\lvert A_C+2\rvert E)$	$(\lvert A_C\rvert+1)E+2\lvert A_C\rvert P$
Our scheme	$(\lvert\mathcal{U}\rvert+2N)E$	$(\lvert A_U\rvert+3\lvert I_U\rvert)E$	$(\lvert A_C\rvert+3)E$	$\lvert A_C\rvert E+(\lvert A_C\rvert+\lvert I_C\rvert+1)P$

Phase 1. The adversary queries secret keys for global identifiers u' with a set of attributes A'_U, where $A_C \not\subseteq A'_U$.

1. For $A_k \in C_{\mathcal{A}}$, it can use $(v_k, \beta_k, w_{k,j})$ to compute secret keys for $a_{k,j} \in \tilde{A}_k \bigcap A'_U$.
2. For $A_k \notin C_{\mathcal{A}}$ and $A_k \neq \mathbf{A}$, it chooses $r_k \xleftarrow{R} \mathbb{Z}_p$ and a random (k_k-1)-degree polynomial $p_k(x) \xleftarrow{R} \mathbb{Z}_p[x]$ with $p_k(0) = r_k$. It computes

$$D_k = B^{v_k} h^{r_k} h_1^{u'\beta_k}.$$

1. If $a_{k,j} \in A_C \bigcap \tilde{A}_k$, it computes

$$D_{k,j} = h^{\frac{p_k(a_{k,j})}{w_{k,j}}}.$$

2. If $a_{k,j} \in \tilde{A}_k - A_C$, it computes

$$D_{k,j} = h^{\frac{p_k(a_{k,j})}{(\alpha+\gamma)w_{k,j}}} = g^{\frac{p_k(a_{k,j})}{w_{k,j}}}.$$

3. For authority $\mathbf{A}$, it chooses $r, e_1, e_2, \ldots, e_{k-1} \xleftarrow{R} \mathbb{Z}_p$, and computes

$$D = B^{-\gamma} h^r \prod_{A_k \notin C_{\mathcal{A}}} B^{-v_k} \prod_{A_k \in C_{\mathcal{A}}} g^{-v_k} h_1^{u'\beta}.$$

a. If $a_j \in A_C \bigcap \tilde{\mathbf{A}}$, it computes

$$D_j = h^{\frac{e_j}{w_j}}.$$

b. If $a_j \in \tilde{\mathbf{A}} - A_C$, it computes

$$D_j = (g^r B^{-1})^{\frac{\Delta_{0,S}(a_j)}{w_j}} \prod_{i=1}^{k-1} g^{\frac{e_i \Delta_{i,S}(a_j)}{w_j}}.$$

We claim that D and D_j are correctly distributed.

$$\begin{aligned} D &= B^{-\gamma} h^r \prod_{A_k \notin C_{\mathcal{A}}} B^{-v_k} \prod_{A_k \in C_{\mathcal{A}}} g^{-v_k} h_1^{u'\beta} \\ &= g^{-b\gamma}(g^a g^\gamma)^r g^{-\left(\sum_{A_k \notin C_{\mathcal{A}}} bv_k + \sum_{A_k \in C_{\mathcal{A}}} v_k\right)} h_1^{u'\beta} \\ &= (g^a g^\gamma)^{-b} g^{ab} (g^a g^\gamma)^r g^{-\left(\sum_{A_k \notin C_{\mathcal{A}}} bv_k + \sum_{A_k \in C_{\mathcal{A}}} v_k\right)} h_1^{u'\beta} \\ &= g^{ab}(g^a g^\gamma)^{r-b} g^{-\left(\sum_{A_k \notin C_{\mathcal{A}}} bv_k + \sum_{A_k \in C_{\mathcal{A}}} v_k\right)} h_1^{u'\beta} \\ &= g^{ab-\left(\sum_{A_k \notin C_{\mathcal{A}}} bv_k + \sum_{A_k \in C_{\mathcal{A}}} v_k\right)} h^{r-b} h_1^{u'\beta}. \end{aligned}$$

Let $r' = r - b$, we have

$$D = g^{ab-\left(\sum_{A_k \notin C_{\mathcal{A}}} bv_k + \sum_{A_k \in C_{\mathcal{A}}} v_k\right)} h^{r'} h_1^{u'\beta}.$$

By selecting $e_1, e_2, \ldots, e_{k-1}$, we implicitly define a $(k-1)$-degree polynomial $p(x) \in \mathbb{Z}_p[x]$, such that $p(0) = r'$ and $p(i) = e_i$. So, we can compute any value of $p(x)$ by interpolation as follows:

$$p(x) = r'\Delta_{0,S}(x) + \sum_{i=1}^{k-1} e_i \Delta_{i,S}(x),$$

where $S = (A_C \bigcap \tilde{\mathbf{A}}) \bigcup \{0\}$.

Hence, for $a_j \in \tilde{\mathbf{A}} - A_C$,

$$\begin{aligned} D_j &= h^{\frac{p(a_j)}{(\alpha+\gamma)w_j}} = g^{\frac{p(a_j)}{w_j}} = g^{\frac{r'\Delta_{0,S}(a_j)}{w_j}} \prod_{i=1}^{k-1} g^{\frac{e_i\Delta_{i,S}(a_j)}{w_j}} \\ &= (g^r B^{-1})^{\frac{\Delta_{0,S}(a_j)}{w_j}} \prod_{i=1}^{k-1} g^{\frac{e_i\Delta_{i,S}(a_j)}{w_j}}. \end{aligned}$$

Challenge. $\mathcal{A}$ sends $\mathcal{B}$ two messages M_0 and M_1 with equal length. $\mathcal{B}$ flips an unbiased coin with $\{0,1\}$, and obtains $\hat{\mu} \in \{0,1\}$. $\mathcal{B}$ computes

$$C_1 = Z \cdot M_{\hat{\mu}}, C_2 = C, C_{i,j} = \{C^{w_{i,j}}\}_{a_{i,j} \in A_C}.$$

$\mathcal{B}$ responds with the challenged ciphertext $(C_1, C_2, \{C_{i,j}\}_{a_{i,j} \in A_C})$. So $(C_1, C_2, \{C_{i,j}\}_{a_{i,j} \in A_C})$ is a valid encryption of $M_{\hat{\mu}}$ with correct distribution whenever $Z = e(g,g)^{abc}$.

Phase 2. Phase 1 is repeated.

Guess. $\mathcal{A}$ outputs his guess $\tilde{\mu}$ on $\hat{\mu}$. If $\tilde{\mu} = \hat{\mu}$, $\mathcal{B}$ outputs his guess $\mu' = 0$. Otherwise, $\mathcal{B}$ outputs his guess $\mu' = 1$.

As shown above, the public parameters and the secret keys generated in the simulation paradigm are identical to those of the real protocol. Now, we compute the probability with which $\mathcal{B}$ can break the BDDH assumption.

If $\mu = 0$, $(C_1, C_2, \{C_{i,j}\}_{a_{i,j} \in A_C})$ is the correct ciphertext of $M_{\hat{\mu}}$. Thereafter, $\mathcal{A}$ can output $\tilde{\mu} = \hat{\mu}$ with advantage at least $\epsilon(\ell)$, namely $Pr[\tilde{\mu} = \hat{\mu} | \mu = 0] \geq \frac{1}{2} + \epsilon(\ell)$. Since $\mathcal{B}$ outputs $\mu' = 0$ when $\tilde{\mu} = \hat{\mu}$, we have $Pr[\mu' = \mu | \mu = 0] \geq \frac{1}{2} + \epsilon(\ell)$

If $\mu = 1$, $\mathcal{A}$ cannot get any information about $\hat{\mu}$. Therefore, $\mathcal{A}$ can output $\tilde{\mu} \neq \hat{\mu}$ with no advantage, namely $Pr[\tilde{\mu} \neq \hat{\mu} | \mu = 1] = \frac{1}{2}$. Since $\mathcal{B}$ outputs $\mu' = 1$ when $\tilde{\mu} \neq \hat{\mu}$, we have $Pr[\mu' = \mu | \mu = 1] = \frac{1}{2}$.

Therefore, the advantage with which $\mathcal{B}$ can break the BDDH assumption is $|\frac{1}{2}Pr[\mu' = \mu | \mu = 0] + \frac{1}{2}Pr[\mu' = \mu | \mu = 1] - \frac{1}{2}| \geq \frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \epsilon(\ell) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} \geq \frac{1}{2}\epsilon(\ell)$. □

We compare our scheme with other multiauthority schemes in Tables 1 and 2. By $|\mathcal{U}|$, $|A_U|$, and $|A_C|$, we denote the number of the universal attributes, the attributes held by user U, and the attributes required by the ciphertext,

TABLE 2
The Comparison of Type, Central Authority, Security Model and the Length of Ciphertext

Schemes	KP/CP-ABE	Central Authority	Security Model	Length of Ciphertext
Chase's scheme [5]	KP-ABE	Yes	Selective-set	$(\lvert A_C\rvert+1)E_{\mathbb{G}}+E_{\mathbb{G}_\tau}$
MKE's [6], [31]	CP-ABE	Yes	Full security	$2\lvert I_C\rvert E_{\mathbb{G}}+\lvert I_C\rvert E_{\mathbb{E}_\tau}$
CC scheme [15]	KP-ABE	No	Selective-set	$(\lvert A_C\rvert+1)E_{\mathbb{G}}+E_{\mathbb{G}_\tau}$
LW's scheme [8]	CP-ABE	No	Full security	$2\lvert A_C\rvert E_{\mathbb{G}}+(\lvert A_C\rvert+1)E_{\mathbb{G}_\tau}$
LCHWY scheme [32]	CP-ABE	Multiple	full security	$(2\lvert A_C\rvert+1)E_{\mathbb{G}}+E_{\mathbb{G}_\tau}$
Our scheme	KP-ABE	No	Selective-set	$(\lvert A_C\rvert+2)E_{\mathbb{G}}+E_{\mathbb{G}_\tau}$

$U(params, PK_i, u)$ $A_i(params, PK_i, SK_i, \mathbb{A}_i)$

1. Chooses $z, \rho_1 \overset{R}{\leftarrow} \mathbb{Z}_p$, and Computes $T = g^z h_1^u$, $P_1 = h^{\rho_1}$. Let $\sum_1$ denote the proof: $PoK\{(z, u, \rho_1) : T = g^z h_1^u \wedge P_1 = h^{\rho_1}\}$

$\xrightarrow{T, P_1, \sum_1}$

2. Chooses $r_i, \rho_2 \overset{R}{\leftarrow} \mathbb{Z}_p$ and a random $(k_i - 1)$-degree polynomial $p_i(x) \overset{R}{\leftarrow} \mathbb{Z}_p[x]$ with $p_i(0) = r_i$, and computes $P_2 = h^{\rho_2}$, $\tilde{D}_i = g^{\alpha_i}(P_1P_2)^{r_i}T^{\beta_i}$, $D^1_{i,j} = (P_2)^{\frac{p_i(a_{i,j})}{t_{i,j}}}$, and $D^2_{i,j} = h^{\frac{p_i(a_{i,j})}{t_{i,j}}}$. Let $\sum_2$ denote the proof:

$PoK\{(\alpha_i, \beta_i, r_i, \rho_2) : \tilde{D}_i = g^{\alpha_i}(P_1P_2)^{r_i}T^{\beta_i} \wedge P_2 = h^{\rho_2} \wedge D^1_{i,j} = (P_2)^{\frac{p_i(a_{i,j})}{t_{i,j}}} \wedge D^2_{i,j} = h^{\frac{p_i(a_{i,j})}{t_{i,j}}} \wedge \prod_{a_{i,j}\in A^i_U} e(T_{i,j}, D^2_{i,j})^{\Delta_{a_{i,j}, A^i_U}(0)} = e(g,h)^{r_i}\}$

$\xleftarrow[D^1_{i,j},\ D^2_{i,j}]{P_2, \tilde{D}_i, \sum_2}$

3. Computes $D_i = \frac{\tilde{D}_i}{Z_i^z}$ and $D_{i,j} = D^1_{i,j}(D^2_{i,j})^{\rho_1}$, where $Z_i = g^{\beta_i}$

Fig. 2. A privacy-preserving key extract protocol BlindKeyGen for the DKP-ABE scheme described in Fig. 1.

respectively. I_U and I_C denote the index set of the authorities such that $A^i_U \neq \{\phi\}$ and $A^i_C \neq \{\phi\}$, respectively. By E and P, we denote one exponential and one paring operation, respectively. By $E_{\mathbb{G}}$ and $E_{\mathbb{G}_\tau}$, we denote one element in group $\mathbb{G}$ and one element in group $\mathbb{G}_\tau$, respectively. N denotes the number of the authorities in the systems. By d, we denote the number of the central authorities in [32].

Collusion Resistance. To be secure against the collusion attacks, the user's identifier u is embedded in his secret keys and bound with the second secret keys of the authorities so that these keys can be tied together. When encrypting a message, all the second public keys of the authorities A_i with $i \in I_C$ are aggregated and randomized by the value s. Therefore, only the secret keys from the same identifier can be used to decrypt the ciphertext. The secret keys from different identifiers cannot be combined as C_3 cannot be split by the malicious users. Suppose that $I_C = I_{C'} \bigcup I_{C''}$ and two users U_1 and U_2 obtain secret keys for the attributes which satisfy the access structures specified by the authorities with the indexes in $I_{C'}$ and $I_{C''}$, respectively. If they cooperate to decrypt the ciphertext, they must compute $C'_3 = \prod_{i\in I_{C'}} g^{\beta_i s}$ and $C''_3 = \prod_{i\in I_{C''}} g^{\beta_i s}$. However, both C'_3 and C''_3 cannot be obtained from C_3 as the exponent s is unknown.

Fine-grained access control. We can only express a threshold access structure in the proposed decentralized KP-ABE scheme above. In order to express any access structure, we employ the *access tree* technique introduced by Goyal et al. [16]. Let $\mathcal{T}$ be a tree which specifies an access structure, and defines an ordering between the children of every node x from 1 to n_x, where n_x denotes the number of the children of the node x. Each nonleaf node of $\mathcal{T}$ represents a threshold gate which consist of the number of its children and a threshold value. Let k_x be the threshold value in the node x, where $0 < k_x \leq n_x$. When $k_x = 1$, the threshold gate is an OR gate. If $k_x = n_x$, the gate is an AND gate. Each leaf node in $\mathcal{T}$ is labeled with an attribute and a threshold value $k_x = 1$. Given an access structure, a polynomial p_x is selected for each node in $\mathcal{T}$ following the way in a top-down manner. Starting from the root node r, set the degree d_x of the polynomial to be $k_x - 1$. In our case, we can set $p_r(0) = r_i$ for the authority A_i. For other nodes in $\mathcal{T}$, we can set $q_x(0) = q_{parent(x)}(index(x))$, where $parent(x)$ denotes the parent node of x, and $index(x)$ denotes the number associated with the node x.

3.2 Privacy-Preserving Key Extract Protocol

We propose a privacy-preserving key extract protocol for the proposed decentralized KP-ABE in Fig. 2.

Overview. In Fig. 1, the secret keys for the user with identifier u are $D_i = g^{\alpha_i}h^{r_i}h_1^{u\beta_i}$ and

$$\{D_{i,j} = h^{\frac{p_i(a_{i,j})}{t_{i,j}}}\}_{a_{i,j}\in A^i_U}.$$

To obtain secret keys from the authority A_i blindly, the user needs to prove that he holds the identifier u in zero-knowledge. Notably, if the random number r_i is chosen by A_i, he can detect the user by computing $h_1^u = (\frac{D_i}{g^{\alpha_i} h^{r_i}})^{\beta_i^{-1}}$, where the identifer u is public. Hence, the random number used to generate secret keys for the user should be computed by executing a 2-party secure computing between the user and A_i. As a result, the authority generates secret keys for the user's attributes without knowing his identifier.

The user U chooses $z, \rho_1 \xleftarrow{R} \mathbb{Z}_p$ and computes $T = g^z h_1^u$ and $P_1 = h^{\rho_1}$. Indeed, T is the commitment of the identifier u and can be used to prove that u has been included in it in zero-knowledge. P_1 will be used to execute 2-party secure computing with A_i. The user proves that he knows z, u, ρ_1 to A_i in zero-knowledge.

If the proof is correct, A_i selects $r_i, \rho_2 \xleftarrow{R} \mathbb{Z}_p$ and a $(k_i - 1)$-degree polynomial $p_i(x) \xleftarrow{R} \mathbb{Z}_p[x]$ with $p_i(0) = r_i$. A_i computes

$$P_2 = h^{\rho_2}, \tilde{D}_i = g^{\alpha_i}(P_1P_2)^{r_i} T^{\beta_i}$$
$$= g^{\alpha_i} h^{r_i(\rho_1+\rho_2)} h_1^{u\beta_i} g^{z\beta_i}, D_{i,j}^1 = (P_2)^{\frac{p_i(a_{i,j})}{t_{i,j}}},$$

and $D_{i,j}^2 = h^{\frac{p_i(a_{i,j})}{t_{i,j}}}$. Indeed, P_1 and P_2 are used to compute the exponential $r_i(\rho_1 + \rho_2)$ by executing a 2-party secure computing. In this case, the secret key for attribute $a_{i,j} \in \tilde{A}_i$ should be

$$D_{i,j} = h^{\frac{(\rho_1+\rho_2)p_i(a_{i,j})}{t_{i,j}}}.$$

Unfortunately, A_i does not know ρ_1. Therefore, he computes

$$D_{i,j}^1 = (P_2)^{\frac{p_i(a_{i,j})}{t_{i,j}}} = h^{\frac{\rho_2 p_i(a_{i,j})}{t_{i,j}}}$$

and $D_{i,j}^2 = h^{\frac{p_i(a_{i,j})}{t_{i,j}}}$ so that the user can compute $D_{i,j}$ from $D_{i,j}^1$, $D_{i,j}^2$, and ρ_1. A_i sends $(P_2, \tilde{D}_i, \{D_{i,j}^1, D_{i,j}^2\}_{a_{i,j} \in A_U^i})$ to the user and proves that he knows $(r_i, \rho_2, \alpha_i, p_i(x), \{t_{i,j}\}_{a_{i,j} \in A_U^i})$ in zero-knowledge.

If the proof is correct, the user can compute his secret keys as $D_i = \frac{\tilde{D}_i}{Z_i^z} = g^{\alpha_i} h^{r_i(\rho_1+\rho_2)} h_1^{u\beta_i}$ and

$$D_{i,j} = D_{i,j}^1 (D_{i,j}^2)^{\rho_1} = h^{\frac{(\rho_1+\rho_2)p_i(a_{i,j})}{t_{i,j}}}.$$

In the BlindKeyGen protocol, the user obtains his secret key $SK_U^i = (D_i, D_{i,j})$ from the authority A_i, where $D_i = g^{\alpha_i} h^{r_i(\rho_1+\rho_2)} h_1^{u\beta_i}$ and

$$D_{i,j} = h^{\frac{p_i(a_{i,j})(\rho_1+\rho_2)}{t_{i,j}}}.$$

The value $r_i(\rho_1 + \rho_2)$ is computed by U and A_i executing 2-party secure computing, where α_i, β_i, r_i, and ρ_2 are from A_i and ρ_1 is from U. Hence, from the view of A_i, $(D_i, D_{i,j})$ is identically distributed in the group $\mathbb{G}$.

The protocol in Fig. 2 works as follows:

1. The user U chooses $\rho_1, z, z_1, z_2, z_3 \xleftarrow{R} \mathbb{Z}_p$, and computes $T = g^z h_1^u$, $P_1 = h^{\rho_1}$, $T' = g^{z_1} h_1^{z_2}$ and $P_1' = h^{z_3}$. U sends (T, P_1, T', P_1') to the authority A_i.
2. A_i chooses $c \xleftarrow{R} \mathbb{Z}_p$, and sends c to U.
3. U computes $s_1 = z_1 - cz$, $s_2 = z_2 - cu$, and $s_3 = z_3 - c\rho_1$, and sends (s_1, s_2, s_3) to A_i.
4. A_i checks $T' \stackrel{?}{=} g^{s_1} h_1^{s_2} T^c$ and $P_1' \stackrel{?}{=} h^{s_3} P_1^c$. If so, A_i chooses $r_i, \rho_2, w, b_1, b_2, b_3, d_j \xleftarrow{R} \mathbb{Z}_p$ and a random $(k_i - 1)$-degree polynomial $p_i(x)$ with $p_i(0) = r_i$, and computes

$$P_2 = h^{\rho_2}, P_2' = h^w, \tilde{D}_i = g^{\alpha_i}(P_1P_2)^{r_i} T^{\beta_i}, D_{i,j}^1$$
$$= P_2^{\frac{p_i(a_{i,j})}{t_{i,j}}}, D_{i,j}^2 = h^{\frac{p_i(a_{i,j})}{t_{i,j}}}, \tilde{D}_i' = g^{b_1}(P_1P_2)^{b_2} T^{b_3}, Z_i'$$
$$= g^{b_3}, Z' = e(g,h)^{b_2}, V_j^1 = P_2^{d_j},$$

   and $V_j^2 = h^{d_j}$. A_i sends $(D_{i,j}^1, D_{i,j}^2,\ P_2, P_2', Z_i', Z', \tilde{D}_i, \tilde{D}_i', V_j^1, V_j^2)$ to U. Otherwise, aborts.
5. U chooses $c' \xleftarrow{R} \mathbb{Z}_p$, and sends c' to A_i.
6. A_i computes $\gamma_1 = b_1 - c'\alpha_i$, $\gamma_2 = b_2 - c'r_i$, $\gamma_3 = b_3 - c'\beta_i$, $\gamma_4 = w - c'\rho_2$, and $\eta_j = t_j - c' \frac{p_i(a_{i,j})}{t_{i,j}}$. A_i sends $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \eta_j)$ to U.
7. U computes $Z = \prod_{a_{i,j} \in A_U^i} e(T_{i,j}, D_{i,j}^2)^{\Delta_{a_{i,j}, A_U^i}(0)}$, and checks

$$P_2' \stackrel{?}{=} h^{\gamma_4} P_2^{c'}, Z' \stackrel{?}{=} e(g,h)^{\gamma_2} Z^{c'}, V_j^1 \stackrel{?}{=} P_2^{\eta_j}(D_{i,j}^1)^{c'}, V_j^2$$
$$\stackrel{?}{=} h^{\eta_j}(D_{i,j}^2)^{c'}$$

   and $\tilde{D}_i' \stackrel{?}{=} g^{\gamma_1}(P_1P_2)^{\gamma_2} T^{\gamma_3} \tilde{D}_i^{c'}$. If so, U computes $D_i = \frac{\tilde{D}_i}{Z_i^z}$ and $D_{i,j} = D_{i,j}^1 (D_{i,j}^2)^{\rho_1}$. Otherwise, aborts.

Theorem 2. *The proposed privacy-preserving key extract protocol BlindKeyGen in Fig. 2 is both leak-free and selective-failure blind.*

Proof. **Leak freeness.** Suppose that there exists an adversary $\mathcal{U}$ in the real experiment (where $\mathcal{U}$ is interacting with an honest authority A_i running the BlindKeyGen protocol), there will exist a simulator $\hat{\mathcal{U}}$ in the ideal experiment (where $\hat{\mathcal{U}}$ can access the trusted party running the ideal KeyGen protocol) such that no efficient distinguisher $\mathcal{D}$ can distinguish the real experiment from the ideal experiment. The simulator $\hat{\mathcal{U}}$ simulates the communication between the distinguisher $\mathcal{D}$ and the adversary $\mathcal{U}$ by passing the input of $\mathcal{D}$ to $\mathcal{U}$ and the output of $\mathcal{U}$ to $\mathcal{D}$. $\hat{\mathcal{U}}$ works as follows:

1. $\hat{\mathcal{U}}$ sends the adversary $\mathcal{U}$ the public key PK_i of A_i.
2. $\mathcal{U}$ must submit two values T and P_1, and prove $PoK\{(z, u, \rho_1) : T = g^z h_1^u \wedge P_1 = h^{\rho_1}\}$. If the proof fails, $\hat{\mathcal{U}}$ aborts the simulation. Otherwise, by using the rewind technique, $\hat{\mathcal{U}}$ can obtain (z, u, ρ_1).
3. $\hat{\mathcal{U}}$ sends u to the trusted party. The trusted party runs KeyGen to generates $(D_i, D_{i,j})$, and responds it to $\hat{\mathcal{U}}$.
4. $\hat{\mathcal{U}}$ chooses $\lambda \xleftarrow{R} \mathbb{Z}_p$, and computes $\rho_2 = \lambda - \rho_1$, $P_2 = h^{\rho_2}$, $\tilde{D}_i = D_i Z_i^z$, $D_{i,j}^1 = (D_{i,j})^{\frac{\rho_2}{\lambda}}$ and $D_{i,j}^2 = D_{i,j}^{\frac{1}{\lambda}}$. $\hat{\mathcal{U}}$ returns $(P_2, \tilde{D}_i, D_{i,j}^1, D_{i,j}^2)$ to $\mathcal{U}$.

If $(D_i, D_{i,j})$ are correct keys from the trusted party in the ideal experiment, $(P_2, \tilde{D}_i, D_{i,j}^1, D_{i,j}^2)$ are the correct keys from A_i in the real experiment. So, $(D_i, D_{i,j})$ and $(P_2, \tilde{D}_i, D_{i,j}^1, D_{i,j}^2)$ are distributed identically. Therefore, no efficient distinguisher $\mathcal{D}$ can distinguish the real game from the ideal game.

Selective-failure blindness. The adversary $\mathcal{A}_i$ submits the public key PK_i, and two global identifiers u_0 and u_1. Then, a bit $b \in \{0, 1\}$ is randomly selected. $\mathcal{A}_i$ can have a

TABLE 3
The Computing Cost and Communication Cost of the Privacy-Preserving Key Extract Protocol

Scheme	Computation Cost		Communication Cost	
	U	A_i	$U \rightarrow A_i$	$U \leftarrow A_i$
Our scheme	$(14+5\lvert A_U^{\imath}\rvert)e+\lvert A_U\rvert p$	$(15+4\lvert A_U^{\imath}\rvert)e$	$4E_{\mathbb{G}}+4E_{\mathbb{Z}_p}$	$(5+\lvert A_U^{\imath}\rvert)E_{\mathbb{Z}_p}+(5+4\lvert A_U^{\imath}\rvert)E_{\mathbb{G}}+E_{\mathbb{G}_\tau}$

black-box access to $U(params, PK_i, u_b)$ and $U(params, PK_i, u_{1-b})$. Then, U executes BlindKeyGen protocol with $\mathcal{A}_i$, where $\mathcal{A}_i$ plays the role of authority A_i. U outputs secret keys SK_U^b and SK_U^{1-b} for global identifiers u_b and u_{1-b}, respectively. If $SK_U^b \neq \perp$ and $SK_U^{1-b} \neq \perp$, $\mathcal{A}_i$ is given (SK_U^b, SK_U^{1-b}). If $SK_U^b \neq \perp$ and $SK_U^{1-b} = \perp$, $\mathcal{A}_i$ is given $(\epsilon, \perp)$. If $SK_U^b = \perp$ and $SK_U^{1-b} \neq \perp, \mathcal{A}_i$ is given $(\perp, \epsilon)$. If $SK_U^b = \perp$ and $SK_U^{1-b} = \perp, \mathcal{A}_i$ is given $(\perp, \perp)$. At the end, $\mathcal{A}_i$ outputs his prediction b' on b.

In the BlindKeyGen protocol, U sends $\mathcal{A}_i$ two random values $T, P_1 \in \mathbb{G}$, and proves $PoK\{(z, u_b, \rho_1) : T = g^z h_1^{u_b} \wedge P_1 = h^{\rho_1}\}$. Supposed that $\mathcal{A}_i$ runs one or both of the oracles up to this point. Now, it is $\mathcal{A}_i$'s turn to respond. So far, $\mathcal{A}_i$'s view on the two oracles is computationally undistinguishable. Otherwise, the hiding property of the commitment scheme and the witness undistinguishable property of the zero-knowledge proof will be broken. Suppose that $\mathcal{A}_i$ uses any computing strategy to output secret keys $\{\tilde{D}_i, D_{i,j}\}$ for the first oracle. In the following, we will show that $\mathcal{A}_i$ can predict SK_U^b of U without interaction with the two oracles:

1. $\mathcal{A}_i$ checks
$$PoK\left\{(\alpha_i, \beta_i, r_i, \rho_2) : \tilde{D}_i = g^{\alpha_i}(P_1P_2)^{r_i}T^{\beta_i} \wedge P_2 \right.$$
$$= h^{\rho_2} \wedge D_{i,j}^1 = (P_2)^{\frac{p_i(a_{i,j})}{t_{i,j}}} \wedge D_{i,j}^2 = h^{\frac{p_i(a_{i,j})}{t_{i,j}}}$$
$$\left. \wedge \prod_{a_{i,j} \in A_U^i} e(T_{i,j}, D_{i,j}^2)^{\Delta_{a_{i,j}, A_U^i}(0)} = e(g,h)^{r_i} \right\}.$$
If the proof fails, $\mathcal{A}$ sets $SK_U^0 = \perp$.
2. $\mathcal{A}_i$ generates different $(\tilde{D}_i, D_{i,j})$ for the second oracle and a proof of knowledge:
$$PoK\left\{(\alpha_i, \beta_i, r_i, \rho_2) : \tilde{D}_i = g^{\alpha_i}(P_1P_2)^{r_i}T^{\beta_i} \wedge P_2 \right.$$
$$= h^{\rho_2} \wedge D_{i,j}^1 = (P_2)^{\frac{p_i(a_{i,j})}{t_{i,j}}} \wedge D_{i,j}^2$$
$$\left. = h^{\frac{p_i(a_{i,j})}{t_{i,j}}} \wedge \prod_{a_{i,j} \in A_U^i} e(T_{i,j}, D_{i,j}^2)^{\Delta_{a_{i,j}, A_U^i}(0)} = e(g,h)^{r_i} \right\}.$$
$\mathcal{A}_i$ checks the proof, if it fails, it sets $SK_U^1 = \perp$.
3. Finally, $\mathcal{A}_i$ outputs his predication on (u_0, u_1) with (SK_U^0, SK_U^1), if $SK_U^0 \neq \perp$ and $SK_U^1 \neq \perp$; $(\epsilon, \perp)$, if $SK_U^0 \neq \perp$ and $SK_U^1 = \perp$; $(\perp, \epsilon)$, if $SK_U^0 = \perp$ and $SK_U^1 \neq \perp$; $(\perp, \perp)$, if $SK_U^0 = \perp$ and $SK_U^1 = \perp$.

The predication on (u_0, u_1) is correct, and has the identical distribution with the oracle. Because $\mathcal{A}_i$ performs the same check as the honest U, it outputs the valid keys as U obtains from $\text{BlindKeyGen}(A_i(para\ ms, PK_i, SK_i) \leftrightarrow \mathcal{U}(params, PK_i, u))$ when both checks are valid. Hence, if $\mathcal{A}_i$ can predict the final outputs of the two oracles, the advantage of $\mathcal{A}_i$ in distinguishing $U(params, PK_i, u_b)$ from $U(params, PK_i, u_{1-b})$ is the same without the final outputs. Therefore, the advantage of $\mathcal{A}_i$ should come from the received T, P_1, and the proof $\sum_1 : PoK\{(z, u_b, \rho_1) : T = g^z h_1^{u_b} \wedge P_1 = h^{\rho_1}\}$. From the hiding property of the commitment scheme and witness undistinguishable property of the zero-knowledge proof, $\mathcal{A}_i$ cannot distinguish one from the other with non-negligible advantage. □

Therefore, from Theorem 1 and Theorem 2, we have the following theorem:

Theorem 3. *Our privacy-preserving decentralized attribute-based encryption scheme* $\tilde{\prod} =$ *(Global Setup, Authority Setup, BlindKeyGen, Encryption, Decryption) is secure in the selective-set model under the DBDH assumption.*

Protecting privacy is an important issue in distributed systems. Therefore, our privacy-preserving decentralized KP-ABE scheme can be used as a sound solution to construct privacy-preserving data transfer and access control schemes in distributed systems, where the data owner can encrypted his data under a set of attributes so that only the users whose attributes satisfy the specified access structure can access the data. Users can obtain secret keys from multiple parties without being traced and exposing their identities to the authorities. As a result, our scheme can capture the following important properties: 1) Fine-grained access control. Our scheme can implement any access structure using the access tree technique; 2) Improving privacy and security. Users cannot be impersonated as they can obtain secret keys from multiple parties without exposing their identities to them; 3) Efficiency. Multiple authorities can work independently without any cooperation.

We list the computing cost and communication cost of the privacy-preserving key extract protocol in Table 3.

4 Conclusion

The decentralized ABE scheme has attracted a lot of attention, because it can reduce the trust on merely a single centralized authority. In order to resist the collusion attacks in the decentralized ABE schemes, the global identifier GID is used to tie all the user's secret keys from multiple authorities together. However, this will risk the user being traced and impersonated by the corrupted authorities. In this paper, we proposed a privacy-preserving decentralized ABE scheme to protect the user's privacy. In our scheme, all the user's secret keys are tied to his identifier to resist the collusion attacks while the multiple authorities cannot know anything about the user's identifier. Notably, each authority can join or leave the system freely without the need of reinitializing the system and there is no central authority. Furthermore, any access structure can be expressed in our scheme using the access tree technique. Finally, our scheme relies on the standard complexity assumption (e.g., DBDH), rather than the nonstandard complexity assumptions (e.g., DDHI).

Acknowledgments

Jinguang Han was supported by PhD scholarships of Smart Services Cooperative Research Centre (CRC) and University of Wollongong.

References

[1] B.C. Neuman and T. Ts'o, "Kerberos: An Authentication Sewice for Computer Networks," *IEEE Comm. Magazine,* vol. 32, no. 9, pp. 33-38, Sept. 1994.

[2] N.P. Smart, "Access Control Using Pairing Based Cryptography," *CT-RSA '03: Proc. RSA Conf. The Cryptographers' Track,* pp. 111-121, 2003.

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption," *Proc. IEEE Symp. Security and Privacy (S&P '07),* pp. 321-34, May 2007.

[4] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," *EUROCRYPT '05: Proc. Advances in Cryptology,* R. Cramer, ed., pp. 457-473, May 2005.

[5] M. Chase, "Multi-Authority Attribute Based Encryption," *Proc. Theory of Cryptography Conf. (TCC '07),* S.P. Vadhan, ed., pp. 515-534, Feb. 2007.

[6] S. Müller, S. Katzenbeisser, and C. Eckert, "Distributed Attribute-Based Encryption," *Proc. 11th Int'l Conf. Information Security and Cryptology (ICISC '08),* P.J. Lee and J.H. Cheon, eds., pp. 20-36, Dec. 2008.

[7] H. Lin, Z. Cao, X. Liang, and J. Shao, "Secure Threshold Multi-Authority Attribute Based Encryption without a Central Authority," *INDOCRYPT '08: Proc. Int'l Conf. Cryptology in India,* D.R. Chowdhury, V. Rijmen, and A. Das, eds., pp. 426-436, Dec. 2008.

[8] A. Lewko and B. Waters, "Decentralizing Attribute - Based Encryption," *EUROCRYPT '11: Proc. 30th Ann. Int'l Conf. Theory and Applications of Cryptographic Techniques: Advances in Cryptology,* K.G. Paterson, ed., pp. 568-588, May 2011.

[9] J. Li, Q. Huang, X. Chen, S.S.M. Chow, D.S. Wong, and D. Xie, "Multi-Authority Ciphertext-Policy Attribute-Based Encryption with Accountability," *Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS '11),* pp. 386-390, 2011.

[10] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," *CRYPTO '01: Proc. Advances in Cryptology,* J. Kilian, ed., pp. 213-229, Aug. 2001.

[11] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," *CRYPTO '84: Proc. Advances in Cryptology,* G.R. Blakley and D. Chaum, eds., pp. 47-53, Aug. 1985.

[12] C. Gentry, "Practical Identity-Based Encryption without Random Oracles," *EUROCRYPT '06: Proc. Advances in Cryptology,* S. Vaudenay, ed., pp. 445-464, May/June 2006.

[13] B. Waters, "Efficient Identity-Based Encryption without Random Oracles," *EUROCRYPT '05: Proc. Advances in Cryptology,* R. Cramer, ed., pp. 114-127, May 2005.

[14] D. Boneh and X. Boyen, "Efficient Selective-Id Secure Identity-Based Encryption without Random Oracles," *EUROCRYP '04: Proc. Advances in Cryptology,* C. Cachin and J. Camenisch, eds., pp. 223-238, May 2004.

[15] M. Chase and S.S. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," *Proc. ACM Conf. Computer and Comm. Security (CCS '09),* E. Al-Shaer, S. Jha, and A.D. Keromytis, eds., pp. 121-130, Nov. 2009.

[16] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," *Proc. ACM Conf. Computer and Comm. Security (CCS '06),* A. Juels, R.N. Wright, and S.D.C. di Vimercati, eds., pp. 89-98, Oct./Nov. 2006.

[17] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute- Based Encryption with Non-Monotonic Access Structures," *Proc. ACM Conf. Computer and Comm. Security (CCS '07),* P. Ning, S.D.C. di Vimercati, and P.F. Syverson, eds., pp. 195-203, Oct. 2007.

[18] L. Cheung and C. Newport, "Provably Secure Ciphertext Policy Abe," *Proc. ACM Conf. Computer and Comm. Security (CCS '07),* P. Ning, S.D.C. di Vimercati, and P.F. Syverson, eds., pp. 456-465, Oct. 2007.

[19] J. Herranz, F. Laguillaumie, and C. Ráfols, "Constant Size Ciphertexts in Threshold Attribute-Based Encryption," *Proc. Public Key Cryptography (PKC '10),* P.Q. Nguyen and D. Pointcheval, eds., pp. 19-34, May 2010.

[20] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully Secure Functional Encryption: Attribute- Based Encryption and (Hierarchical) Inner Product Encryption," *EUROCRYPT '10: Proc. Advances in Cryptology,* H. Gilbert, ed., pp. 62-91, May/June 2010.

[21] B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization," *Proc. 14th Int'l Conf. Practice and Theory in Public Key Cryptography Conf. Public Key Cryptography (PKC '11),* D. Catalano, N. Fazio, R. Gennaro, and A. Nicolosi, eds., pp. 53-70, Mar. 6-9 2011.

[22] A. Beimel, "Secure Schemes for Secret Sharing and Key Distribution," Phd thesis, Israel Inst. of Technology, Technion, Haifa, Israel, June 1996.

[23] N. Attrapadung and H. Imai, "Dual-Policy Attribute Based Encryption," *Proc. Seventh Int'l Conf. Applied Cryptography and Network Security (ACNS '09),* M. Abdalla, D. Pointcheval, P.-A. Fouque, and D. Vergnaud, eds., pp. 168-185, June 2009.

[24] A. Rial and B. Preneel, "Blind Attribute-Based Encryption and Oblivious Transfer with Fine-Grained Access Control," *Proc. 2010th Benelux Workshop Information and System Security (WISSec '10),* pp. 1-20, 2010.

[25] S. Yu, K. Ren, and W. Lou, "FDAC: Toward Fine-Grained Data Access Control in Wireless Sensor Networks," *IEEE Trans. Parallel and Distributed Systems,* vol. 22, no. 4, pp. 673-686, Apr. 2011.

[26] J. Hur and D.K. Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems," *IEEE Trans. Parallel and Distributed Systems,* vol. 22, no. 7, pp. 1214-1221, July 2011.

[27] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing," *Proc. IEEE INFOCOM '10,* pp. 534-542, Mar. 2010.

[28] R. Gennaro, S.law Jarecki, H. Krawczyk, and T. Rabin, "Secure Distributed Key Generation for Discrete-Log Based Cryptosystems," *EUROCRYPT '99: Proc. 17th Int'l Conf. Theory and Application of Cryptographic Techniques,* J. Stern, ed., pp. 295-310, May 1999.

[29] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Robust Threshold DSS Signatures," *Information and Computation,* vol. 164, no. 1, pp. 54-84, 2001.

[30] M. Naor, B. Pinkas, and O. Reingold, "Distributed Pseudo - Random Functions and KDCs," *EUROCRYPT '99: Proc. Advances in Cryptology,* J. Stern, ed., pp. 327-346, May 1999.

[31] S. Müller, S. Katzenbeisser, and C. Eckert, "On Multi-Authority Ciphertext-Policy Attribute-Based Encryption," *Bull. of the Korean Math. Soc.,* vol. 46, no. 4, pp. 803-819, 2009.

[32] Z. Liu, Z. Cao, Q. Huang, D.S. Wong, and T.H. Yuen, "Fully Secure Multi-Authority Ciphertext-Policy Attribute-Based Encryption without Random Oracles," *Proc. 16th European Symp. Research in Computer Security (ESORICS '11),* V. Atluri and C. Diaz, eds., pp. 278-297, Sept. 2011.

[33] T.P. Pedersen, "Non-Interactive and Information- Theoretic Secure Verifiable Secret Sharing," *CRYPTO '91: Proc. 11th Ann. Int'l Cryptology Conf. Advances in Cryptology,* J. Feigenbaum, ed., pp. 129-140, Aug. 1991.

[34] J. Camenisch and M. Stadler, "Efficient Group Signature Schemes for Large Groups," *CRYPTO '97: Proc. 17th Ann. Int'l Cryptology Conf. Advances in Cryptology,* B.S. Kaliski, Jr., ed., pp. 410-424, Aug. 1997.

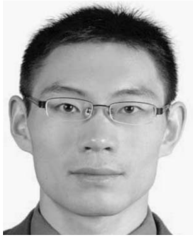
[35] J. Camenisch, M. Kohlweiss, A. Rial, and C. Sheedy, "Blind and Anonymous Identity-Based Encryption and Authorised Private Searches on Public Key Encrypted Data," *Proc. 12th Int'l Conf. Practice nad Theory in Public Key Cryptography (PKC '09),* S. Jarecki and G. Tsudik, eds., pp. 196-214, Mar. 2009.

[36] M. Green and S. Hohenberger, "Blind Identity-Based Encryption and Simulatable Oblivious Transfer," *ASIACRYPT '07: Proc. 13th Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology,* K. Kurosawa, ed., pp. 265-282, Dec. 2007.

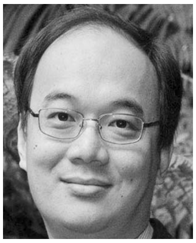
[37] D. Chaum, "Security without Identification: Transaction Systems to Make Big Brother Obsolete," *Comm. ACM,* vol. 28, no. 10, pp. 1030-1044, 1985.

[38] J. Camenisch and A. Lysyanskaya, "An Efficient System for Non-Transferable Anonymous Credentials with Optional Anonymity Revocation," *EUROCRYPT '01: Proc. Int'l Conf. the Theory and Application of Cryptographic Techniques: Advances in Cryptology,* B. Pfitzmann, ed., pp. 93-118, May 2001.

[39] A. Lysyanskaya, R.L. Rivest, A. Sahai, and S. Wolf, "Pseudonym Systems," *Proc. Sixth Ann. Int'l Workshop Selected Areas in Cryptography (SAC '99),* H.M. Heys and C. M. Adams, eds., pp. 184-199, Aug. 1999.



Jinguang Han is currently working toward the PhD degree in School of Computer Science and Software Engineering, University of Wollongong, Australia. He is also a lecturer in College of Sciences, Hohai University, China. His research interests include applied cryptography, identity management, access control and data outsourcing. He is a student member of the IEEE.



Willy Susilo received the PhD degree in computer science from the University of Wollongong, Australia. He is a professor at the School of Computer Science and Software Engineering and the codirector of Centre for Computer and Information Security Research (CCISR) at the University of Wollongong. He is currently holding the prestigious ARC Future Fellow awarded by the Australian Research Council (ARC). His main research interests include cryptography and information security. His main contribution is in the area of digital signature schemes. He has served as a program committee member in dozens of international conferences. He has published numerous publications in the area of digital signature schemes and encryption schemes. He is a senior member of the IEEE.

Yi Mu received the PhD degree from the Australian National University in 1994. He currently is an associate professor, head of School of Computer Science and Software Engineering and the codirector of Centre for Computer and Information Security Research at University of Wollongong, Australia. His current research interests include network security, computer security, and cryptography. He is the editor-in-chief of *International Journal of Applied Cryptography* and serves as an associate editor for nine other international journals. He is a senior member of the IEEE and a member of the IACR.

Jun Yan received the BEng and MEng degrees in computer application Technologies from Southeast University, Nanjing, China, in 1998 and 2001, respectively, and the PhD degree in information technology from Swinburne University of Technology, Melbourne, Australia, in 2004. He is currently a senior lecturer in the School of Information Systems and Technology, University of Wollongong, Wollongong, Australia. His research interests include software technologies, workflow management, service-oriented computing, and agent technologies.