Privacy-Preserving Decentralized Ciphertext-Policy Attribute-Based Encryption with Fully Hidden Access Structure

Huiling Qian, Jiguo Li, and Yichen Zhang

College of Computer and Information Engineering
Hohai University, Nanjing, P.R. China, 210098
lijiguo@hhu.edu.cn

Abstract. To make multi-authority ABE schemes collusion-resistant, a user in the system must be tied with a globally verifiable identifier GID. The drawback of this approach is that it compromises the user's privacy. Malicious authorities can collect user's attributes by tracing the user GID, thus compromises the privacy of the user. The other privacy concern is access structures that sent along with ciphertext in traditional CP-ABE schemes may have sensitive information. In this paper, we propose a multi-authority ABE scheme with fully hidden access structure that authorities can get nothing about user GID when generating and issuing user private keys and access structures are hidden to receivers. We prove the security of our scheme under a standard complexity assumption of decisional bilinear Diffie-Hellman (DBDH) assumption. The access structure we used in our scheme is AND, OR gates on multi-valued attributes.

1 Introduction

In distributed file systems, it allows users to access files from different hosts via network. Thus multiple users can share files and store data. To protect the sensitive data, a complicated access control policy is needed to specify who can access those data. However, traditional access control policies may have some drawbacks, especially in distributed systems. The first drawback is management of user identities. In traditional access control policies, a user identity must be validated by the authority when accessing files or data. So, it can be very hard to manage numerous identities in large distributed file systems. Another drawback is privacy concerns. To overcome these problems and drawbacks, Sahai and Waters [1] introduced the concept of ABE. In this scheme, user's secret key and ciphertext are labeled with a set of attributes, when there is a match between the secret keys and ciphertext, the user can decrypt the message. To share his data, the user can specify an access structure on who can access the data. Therefore, ABE schemes make it possible for users to be validated by descriptive attributes rather than a unique identity. Furthermore, ABE schemes enable one-to-many encryption; one can specify an access structure on who can

S. Qing et al. (Eds.): ICICS 2013, LNCS 8233, pp. 363–372, 2013.

decrypt the data without knowing specific identity. Users whose attributes satisfy the access structure can decrypt the data and access the file.

There are two forms of ABE schemes, key-policy attribute based encryption (KP-ABE) and ciphertext-policy attribute based encryption (CP-ABE). In a KP-ABE scheme [2], secret keys are associated with an access structure and ciphertext is labeled by a set of attributes. If and only if the set of attributes in the ciphertext satisfy the access structure in the secret keys, the user can access the encrypted data. Conversely, in a CP-ABE scheme [3], ciphertext is associated with an encryptor specified access structure and secret keys are labeled by a set of attributes.

1.1 Related Work

The scheme proposed by Sahai and Waters [1] in 2005 can only express simple (t, n) threshold access structure. The limited expressive power is a restriction to the applicability of ABE schemes. Some efforts have been made to enhance the expressibility of ABE schemes. Goyal et al. [2] greatly improved the expressibility of ABE schemes by proposing an ABE scheme with fine-grained access control. Ostrovsky et al. [4] gave the first KP-ABE scheme supporting non-monotonic access structure.

The first CP-ABE scheme is proposed by Bethencourt et al.[3]. In this scheme, it allows the encryptor to specify an access structure in terms of any monotonic access formula. Cheung and Newport [5] constructed a CP-ABE scheme, its complexity assumption is bilinear Diffie-Hellman assumption. However, the scheme only supports positive and negative attributes and wildcards in the access structure. To enhance the expressibility of the access structure, Balu et al. [6] proposed a new CP-ABE scheme, the access structure in this scheme can be expressed by AND, OR gates on multi-valued attributes. In traditional CP-ABE schemes [3,5], access structures are sent to receivers along with ciphertexts. However, access structures may contain some sensitive information. To address this issue, Boneh and Waters [7] proposed a predicate encryption scheme based on hidden vector encryption. Nishide et al. [8] proposed CP-ABE schemes with hidden access structure. In [9], the authors proposed a fully secure CP-ABE with partially hidden access structures.

In all the schemes we discussed above, there is only one authority monitoring and issuing user secret keys. However, there will often be more than one party that acts as authority in reality. Chase [10] proposed the first multi-authority ABE scheme in 2007. In this scheme, there are multiple authorities responsible for monitoring attributes and issuing secret keys. There also exists a central authority generating public and secret keys for other authorities. Users get their secret keys from multiple authorities. Different approaches have been provided to remove the trusted central authority. In [11], a technique named distributed PRF is used to remove the central authority. Moreover, the authors first give the concern that malicious authorities might collect user's attributes and combine their own information to build a full profile, thus compromises the privacy of the user. In [12], the scheme removes the need of cooperation with authorities

in the setup stage. They also remove the need of central authority, thus making the system more scalable.

Multi-authority ABE scheme is more in line with reality, different authorities monitor different sets of attributes. However, being different from single authority ABE scheme, to resist collusion attacks in multi-authority ABE schemes is difficult. Chase [10] solved this problem by introducing global identifier GID. However, this solution compromises user's privacy. Malicious authorities can collaborate and collect user's attributes by tracing user's GID, thus compromises the privacy of the user. Han et al. [13] addressed this issue by involving a 2-party secure computation protocol based on the ideas in [11].

1.2 Our Contributions

In this paper, we propose a decentralized multi-authority CP-ABE scheme. Multiple authorities monitor different kinds of attributes. Moreover, we remove the need of trusted central authority. Even parts of the authorities are not honest, our scheme remains secure. Authorities in our scheme do not need to collaborate in the setup stage. Authorities can join and leave the system freely. In our scheme, the access structure is fully hidden, and authorities in our scheme can get nothing about user GID. Thus we protect user privacy from both malicious users and malicious authorities.

2 Preliminaries

Definition 1. (Bilinear Maps).*Let $\mathbb{G}$, $\mathbb{G}_T$ be two multiplicative cyclic groups of prime order p. Let g be a generator of $\mathbb{G}$ and e be a bilinear map, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. The bilinear map e has the following properties:*

- Bilinearity: for all $g, h \in \mathbb{G}$, and $a, b \in \mathbb{Z}_p$, we have $e(g^a, h^b) = e(g, h)^{ab}$.
- Non-degeneracy: $e(g, g) \neq 1$.
- Computability: Group operation $e(g, h)$ is efficiently computable, where $g, h \in \mathbb{G}$.

Definition 2. (Decisional Bilinear Diffie-Hellman (DBDH) Assumption)[14]. *Let $a, b, c, z \in \mathbb{Z}_p$ be chosen at random and g be a generator of group $\mathbb{G}$. The DBDH assumption holds when no polynomial-time algorithm $\mathcal{B}$ can distinguish the tuple $(A, B, C, Z) = (g^a, g^b, g^c, g^{abc})$ from the tuple $(A, B, C, Z) = (g^a, g^b, g^c, g^z)$ with non-negligible advantage. The advantage of algorithm $\mathcal{B}$ is*

$$\mathtt{Adv}_{\mathcal{B}}^{DBDH} = |Pr[\mathcal{B}(A, B, C, g^{abc}) = 1] - Pr[\mathcal{B}(A, B, C, g^z) = 1]|.$$

Definition 3. (Access Structure)[15]. *Let $P = \{P_1, P_2, ..., P_n\}$ be a set of parties. A collection $\mathbb{A} \subseteq 2^P$ is considered to be monotone if $\forall B, C$ satisfies that if $B \in \mathbb{A}$ and $B \subseteq C$, then $C \in \mathbb{A}$. An access structure (resp., monotonic access structure) is a collection (resp., monotone collection) $\mathbb{A}$ that $\mathbb{A} \subseteq 2^P\{\emptyset\}$. The sets in $\mathbb{A}$ are called the authorized sets, and the sets not in $\mathbb{A}$ are called the unauthorized sets.*

Commitment. A commitment scheme allows someone to commit a chosen value without leaking the value for a period of time and reveal the committed value later when it is needed. The commitment scheme used in our scheme is a perfectly hiding commitment scheme named as Pedersen commitment scheme [16].

Zero-Knowledge Proof. A zero-knowledge proof is an interactive proof for a prover to prove some knowledge without revealing the knowledge. The zero-knowledge proof scheme involved in our construction is introduced by Camenisch and Stadler [17].

3 Formal Definition and Security Model

3.1 Outline of Decentralized CP-ABE Encryption

A decentralized CP-ABE scheme consists of the following five algorithms.

Global Setup: This algorithm takes an implicit security parameter l as input and returns the system parameters params for the system.

Authority Setup: This algorithm is run by authorities in the system. Each authority A_k generates his secret keys SK_k and public keys PK_k, where $k = 1, 2, ..., N$.

KeyGen: This algorithm takes authority's secret keys SK_k, a set of attributes L^k and a global identifier GID as input, returns the secret keys SK_U^k for user U. Here $L^k = \hat{A}_k \cap L$, $\hat{A}_k$ denotes the attributes monitored by the authority A_k, L denotes the list of attributes corresponding to the GID.

Encryption: The encryption algorithm takes the system parameters params, a message M, authority's public keys PK_k and an access structure W as input, returns the ciphertext C_T.

Decryption: This algorithm takes the global identifier GID, a collection of secret keys corresponding to user attributes and the ciphertext C_T as input, and outputs the message M when user attributes satisfy the encryptor specified access structure.

3.2 Security Model

The security game is played between adversary and challenger as follows:

Initialization: Adversary $\mathcal{A}$ submits the challenge access structure W_0^*, W_1^* and a list of corrupted authorities $C_{\mathcal{A}}$ to algorithm $\mathcal{B}$, where $|C_{\mathcal{A}}| < N$.

Global Setup: The challenger runs the algorithm **Setup** and outputs the system parameters params to adversary $\mathcal{A}$.

Authorities Setup: For the corrupted authorities, the challenger sends his public and secret keys (PK_k, SK_k) to the adversary $\mathcal{A}$. For the honest authorities, the challenger sends his public keys PK_k to the adversary $\mathcal{A}$. For the third kind of authorities, the challenger sends his public keys PK_k and parts of secret keys SK_k to the adversary $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ sends an attribute list L to the challenger for secret keys queries, where ($L \nvDash W_0^*$ or $L \nvDash W_1*$) and ($L \nvDash W_0^*$ and $L \nvDash W_1*$). The challenger returns secret keys for these attributes.

Challenge: The adversary $\mathcal{A}$ submits two equal length messages M_0 and M_1. The challenger chooses a random bit $\xi \in_R \{0,1\}$ and runs the algorithm **Encryption**. The challenger gives the ciphertext $C^*_{T,\xi}$ to the adversary $\mathcal{A}$. Note that if $L \nvDash W_0^*$ and $L \nvDash W_1^*$, then $M_0 = M_1$.

Phase 2: Phase 1 is repeated.

Guess: Finally adversary $\mathcal{A}$ outputs his guess ξ' on ξ.

Definition 4. *A decentralized CP-ABE scheme is* (t, q, ϵ) *secure in the selective-set model if all* t*-time adversary makes at most* q *secret key queries and succeeds in the above game with negligible advantage* ϵ*.*

3.3 Outline of Privacy-Preserving Decentralized CP-ABE Encryption

To protect user privacy from malicious authorities, we replace the algorithm **KeyGen** in the decentralized CP-ABE encryption scheme with **BlindKeyGen**. Other algorithms remain the same. The algorithm **BlindKeyGen** is described as follows.

BlindKeyGen: User U runs the algorithm Commit and returns *com* to the authority A_k. Authority A_k uses *com* to verify whether the user U has GID u or not in zero-knowledge. If the proof is correct, authority A_k computes partial secret keys for the user. The user verifies whether the authority A_k has the correct secret keys in zero-knowledge through partial secret keys. If the proof is correct and Decommit returns 1, the user U can compute his secret keys successfully and authority A_k gets empty. Otherwise, algorithm aborts and outputs $(\perp, \perp)$ for the authority and user.

To be secure against both malicious users and malicious authorities, algorithm **BlindKeyGen** should satisfy two properties: leak freeness and selective-failure blindness [18,13]. Leak freeness requires that a malicious user can get nothing by executing algorithm **BlindKeyGen** with an honest authority. Selective-failure blindness requires that a malicious authority cannot get anything about user's GID u and cannot fail the algorithm according to user's GID u through running algorithm **BlindKeyGen**.

4 Our Construction

In this section, we propose a decentralized CP-ABE scheme which can fully hide access structure specified by the encryptor.

4.1 Decentralized CP-ABE Encryption Scheme with Fully Hidden Access Structure

Our scheme is constructed as follows.

Global Setup: Given the security parameter l, the algorithm returns a bilinear group $(e, p, \mathbb{G}, \mathbb{G}_T)$ with prime order p. Let g, h and h_1 be the generators of group $\mathbb{G}$. Suppose there are N authorities in the system, namely $A_1, A_2, ..., A_N$.

Authority Setup: Each authority A_k chooses $\alpha_k \in_R \mathbb{Z}_p, \beta_k \in_R \mathbb{Z}_p$ and $t_{i,j}^k \in_R \mathbb{Z}_p (i \in [1,n], j \in [1,n_i])$, and computes $Y_k = e(g,g)^{\alpha_k}, Z_k = g^{\beta_k}$, and $T_{i,j}^k = g^{t_{i,j}^k}$. The secret keys and public keys of authority A_k are $SK_k = (\alpha_k, \beta_k, \{t_{i,j}^k\}_{i\in[1,n],j\in[1,n_i]})$ and $PK_k = (Y_k, Z_k, \{T_{i,j}^k\}_{i\in[1,n],j\in[1,n_i]})$.

KeyGen: Denote the user's global identifier GID by u, where $u \in \mathbb{Z}_p$. Let L be the attribute list of the user U. To generate a key for the user U, authority A_k selects $r_k, \tau_k \in_R \mathbb{Z}_p$, $\omega_i \in_R \mathbb{Z}_p^*$ for $1 \le i \le n$ and computes

$D_{i,1}^k = g^{\alpha_k} h^{r_k} h_1^{\frac{1}{u+\beta_k}}, D_{i,2}^k = h^{\omega_i t_{i,j}^k}, D_{i,3}^k = h^{\omega_i}, D_0^k = h^{r_k} h_1^{-\tau_k}, D_1^k = h_1^{\tau_k + \frac{1}{u+\beta_k}}$ *for* $t_{i,j}^k \in L^k$, where $L^k = \hat{A}_k \cap L$, for $k = 1, 2, ..., N$, $\hat{A}_k$ denotes the attributes monitored by the authority A_k.

Encryption: An encryptor chooses a random number $s \in_R \mathbb{Z}_p$, and computes

$$C_1 = M \cdot \prod_{k\in I_c} {Y_k}^s, C_2 = g^s,$$

where I_c is an index set of authorities A_k.

The encryptor sets the value of root node to be s, marks the root node as assigned and all the child nodes as un-assigned.

For each non leaf node that is un-assigned, the encryptor proceeds as follows.

1. If the symbol in the access structure is $\wedge$ and its child nodes are un-assigned, the encryptor selects a random number $s_i \in_R \mathbb{Z}_p, 1 \le s_i \le p-1$. For the last child node, set $s_j = s - \sum_{i=1}^{j-1} s_i \bmod p$. Mark this node assigned.
2. If the symbol in the access structure is $\vee$, the encryptor sets the value of this node to be s and mark this node assigned.
3. The encryptor computes $C_{i,j,1} = \prod_{k\in I_c} (T_{i,j}^k)^{s_i}, C_{i,j,2} = g^{s_i}$.

The encryptor outputs the ciphertext $C_T = (C_1, C_2, \{C_{i,j,1}, C_{i,j,2}\}_{i\in[1,n],j\in[1,n_i]})$.

Decryption: To decrypt the ciphertext C_T, the user computes $E = \prod_{k\in I_c} e(D_1^k, C_2), F = \prod_{k\in I_c} e(D_0^k, C_2), P = e(D_{i,3}^k, C_{i,j,1}), Q = \prod_{k\in I_c} e(D_{i,1}^k, C_2), H = \prod_{k\in I_c} e(D_{i,2}^k, C_{i,j,2})$ and $M = C_1 \cdot \frac{PEF}{QH}$.

Now we prove the correctness of our scheme.

$$E = \prod_{k\in I_c} e(D_1^k, C_2) = \prod_{k\in I_c} e(h_1^{\tau_k + \frac{1}{u+\beta_k}}, g^s) = \prod_{k\in I_c} e(g, h_1)^{s(\tau_k + \frac{1}{u+\beta_k})},$$

$$F = \prod_{k\in I_c} e(D_0^k, C_2) = \prod_{k\in I_c} e(h^{r_k} h_1^{-\tau_k}, g^s) = \prod_{k\in I_c} e(g,h)^{sr_k} e(g,h_1)^{-s\tau_k},$$

$$P = e(D_{i,3}^k, C_{i,j,1}) = e(h^{\omega_i}, \prod_{k\in I_c} g^{s_i t_{i,j}^k}) = e(g,h)^{\sum_{k\in I_c} s_i \omega_i t_{i,j}^k},$$

$$H = \prod_{k\in I_c} e(D_{i,2}^k, C_{i,j,2}) = \prod_{k\in I_c} e(g,h)^{\sum_{k\in I_c} s_i \omega_i t_{i,j}^k},$$

$$\begin{aligned} Q &= \prod_{k\in I_c} e(D_{i,1}^k, C_2) = \prod_{k\in I_c} e(g^{\alpha_k} h^{r_k} h_1^{\frac{1}{u+\beta_k}}, g^s) \\ &= \prod_{k\in I_c} e(g,g)^{s\alpha_k} e(g,h)^{sr_k} \prod_{k\in I_c} e(g,h_1)^{\frac{s}{u+\beta_k}}, \end{aligned}$$

$$C_1 \cdot \frac{PEF}{QH} = M \cdot \frac{e(g,h)^{\sum_{k\in I_c} s_i \omega_i t_{i,j}^k} \prod_{k\in I_c} e(g,h_1)^{\frac{s}{u+\beta_k}} e(g,h)^{sr_k} e(g,g)^{s\alpha_k}}{e(g,h)^{\sum_{k\in I_c} s_i \omega_i t_{i,j}^k} \prod_{k\in I_c} e(g,g)^{s\alpha_k} e(g,h)^{sr_k} e(g,h_1)^{\frac{s}{u+\beta_k}}} = M.$$

Theorem 1. *Our decentralized CP-ABE scheme is* (Γ, q, ϵ) *semantically secure in the selective-set model, if the* (Γ', ϵ') *DBDH assumption holds in* $(e, p, \mathbb{G}, \mathbb{G}_T)$*, where*

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

4.2 BlindKeyGen Protocol

The first part of secret keys in the scheme we proposed in section 4.1 is $D_{i,1}^k = g^{\alpha_k} h^{r_k} h_1^{\frac{1}{u+\beta_k}}$. In order to obtain secret keys blindly from authority A_k, the user has to prove his possess of GID u in zero-knowledge. However, if the random number r_k is chosen by authority A_k as the same as we described in section 4.1, then he can compute $h_1^{\frac{1}{u+\beta_k}} = \frac{D_{i,1}^k}{g^{\alpha_k} h^{r_k}}$ or $h_1^{\frac{1}{u+\beta_k}} = \frac{D_1^k}{h_1^{\tau}}$. Since h_1 and u are public, β_k is the part of secret key of authority A_k, authority A_k can identify user GID u by computing $h_1^{\frac{1}{u+\beta_k}}$, which is not allowed according to the property selective-failure blindness of protocol **BlindKeyGen**. Therefore, we use the technique 2-party secure computing to generate the random number r_k and τ_k. The protocol **BlindKeyGen** is described as follows.

1. The user U and authority A_k first use the technique 2-party secure computing to generate $\rho_1(u+\beta_k)$, where ρ_1 is a random number selected by user U. They can operate as follows. Firstly, the user U selects $\rho_1 \in_R \mathbb{Z}_p$, computes $x = u\rho_1$, and returns x to the authority A_k. Secondly, authority A_k selects $\rho_3 \in_R \mathbb{Z}_p$, computes $y = \beta_k \rho_3, x' = \rho_3 x$, and returns (x', y) to the user U. Then, user U computes $y' = \rho_1 y$ and returns y' to authority A_k. Authority A_k computes $X = \frac{x'+y'}{\rho_3}$, and then authority A_k selects $\theta, p_1, x_1, x_2, x_3, x_4 \in_R \mathbb{Z}_p$, computes $T = h_1^{\frac{\theta}{X}}, T_1 = g^{\alpha_k \theta}, P_1 = h^{p_1}, Q_1 = h_1^{p_1}, T' = h_1^{x_1}, T_1' = g^{x_2}, P_1' = h^{x_3}$ and $Q_1' = h_1^{x_4}$ and returns $(T, T_1, P_1, Q_1, T', T_1', P_1', Q_1')$ to the user U.
2. User U selects $c \in_R \mathbb{Z}_p$ and returns c to the authority A_k. Authority A_k computes $a_1 = x_1 - c\frac{\theta}{X}, a_2 = x_2 - c\alpha_k\theta, a_3 = x_3 - cp_1, a_4 = x_4 - cp_1$. Authority A_k returns (a_1, a_2, a_3, a_4) to the user U.
3. User U checks whether $T' = h_1^{a_1} T^c, T_1' = g^{a_2} T_1^c, P_1' = h^{a_3} P_1^c$ and $Q_1' = h^{a_4} Q_1^c$. If the equations hold, user U selects $\rho_2, p_2, y_1, y_2, y_3, y_4, y_5, y_6, y_7 \in_R \mathbb{Z}_p$ and computes $T_2 = (T^{\rho_1} T_1)^{\rho_2}, P_2 = h^{p_2}, Q_2 = h_1^{p_2}, T_3 = T^{\rho_1\rho_2}, P = (P_1 P_2)^{\rho_2}, Q = (Q_1 Q_2)^{\rho_2}, T_2' = T^{y_1} T_1^{y_2}, P_2' = h^{y_3}, Q_2 = h_1^{y_4}, T_3' = T^{y_5}, P' = (P_1 P_2)^{y_6}$ and $Q' = (Q_1 Q_2)^{y_7}$. The user U returns $(T_2, P_2, Q_2, T_3, P, Q, T_2', P_2', Q_2', T_3', P', Q')$ to the authority A_k. The user U should prove his possess of (ρ_2, p_2) to authority A_k in zero-knowledge.
4. Authority A_k selects $c' \in_R \mathbb{Z}_p$ and returns c' to the user U. User U computes $b_1 = y_1 - c'\rho_1\rho_2, b_2 = y_2 - c'\rho_2, b_3 = y_3 - c'p_2, b_4 = y_4 - c'p_2, b_5 = y_5 - c'\rho_1\rho_2, b_6 = y_6 - c'\rho_2, b_7 = y_7 - c'\rho_2$. User U returns $(b_1, b_2, b_3, b_4, b_5, b_6, b_7)$ to the authority A_k.
5. Authority A_k checks whether $T_2' = T^{b_1} T_1^{b_2} T_2^{c'}, P_2' = h^{b_3} P_2^{c'}, Q_2' = h_1^{b_4} Q_2^{c'}, T_3' = T^{b_5} T_3^{c'}, P' = (P_1 P_2)^{b_6} P^{c'}$ and $Q' = (Q_1 Q_2)^{b_7} Q^{c'}$. If the equations hold,

then authority A_k selects $r_k, \tau_k, z_1, z_2, z_3, z_4, z_5, z_6 \in_R \mathbb{Z}_p$, $\omega_i, t_i, \eta_i \in_R \mathbb{Z}_p^*$ for $1 \leq i \leq n$ and computes $\tilde{D}_{i,1}^k = T_2^{\frac{1}{\theta}} P^{r_k}, D_{i,2}^k = h^{\omega_i t_{i,j}^k}, D_{i,3}^k = h^{\omega_i}, \tilde{D}_0^k = P^{r_k} Q^{-\tau_k}, \tilde{D}_1^k = T_3^{\frac{1}{\theta}} Q^{\tau_k}, (\tilde{D}_{i,1}^k)' = T_2^{z_1} P^{z_2}, (D_{i,2}^k)' = (D_{i,3}^k)^{\eta_i}, (D_{i,3}^k)' = h^{t_i}, (\tilde{D}_0^k)' = P^{z_3} Q^{-z_4}, (\tilde{D}_1^k)' = T_3^{z_5} Q^{z_6}$ and returns $(\tilde{D}_{i,1}^k, D_{i,2}^k, D_{i,3}^k, \tilde{D}_0^k, \tilde{D}_1^k, (\tilde{D}_{i,1}^k)', (D_{i,2}^k)', (D_{i,3}^k)', (\tilde{D}_0^k)', (\tilde{D}_1^k)')$ to the user U. Here, we replace the random number r_k and τ_k in the original scheme with $(p_1+p_2)r_k$ and $(p_1+p_2)\tau_k$, where p_1 is only known to authority A_k and p_2 is only known to user U. Thus malicious authority cannot compute $h_1^{\frac{1}{u+\beta_k}}$ and selectively fail the algorithm.

6. User U selects $c'' \in_R \mathbb{Z}_p$ and returns c'' to the authority A_k. Authority A_k computes $c_1 = z_1 - \frac{c''}{\theta}, c_2 = z_2 - c'' r_k, c_3 = z_3 - c'' r_k, c_4 = z_4 - c'' \tau_k, c_5 = z_5 - \frac{c''}{\theta}, c_6 = z_6 - c'' \tau_k, d_i = \eta_i - c' t_{i,j}^k$ and $e_i = t_i - c' \omega_i$ and returns $(c_1, c_2, c_3, c_4, c_5, c_6, d_i, e_i)$ to user U.
7. User U checks whether $(\tilde{D}_{i,1}^k)' = T_2^{c_1} P^{c_2} (\tilde{D}_{i,1}^k)^{c''}, (D_{i,2}^k)' = (D_{i,3}^k)^{d_i} (D_{i,2}^k)^{c''}, (D_{i,3}^k)' = h^{e_i} (D_{i,3}^k)^{c''}, (\tilde{D}_0^k)' = P^{c_3} Q^{-c_4} (\tilde{D}_0^k)^{c''}$ and $(\tilde{D}_1^k)' = T_3^{c_5} Q^{c_6} (\tilde{D}_1^k)^{c''}$ or not. If the equations hold, user U computes $D_{i,1}^k = (\tilde{D}_{i,1}^k)^{\frac{1}{\rho 2}}, D_0^k = (\tilde{D}_0^k)^{\frac{1}{\rho 2}}$ and $D_1^k = (\tilde{D}_1^k)^{\frac{1}{\rho 2}}$. Otherwise, the algorithm aborts.

Theorem 2. *Our **BlindKeyGen** protocol is leak-free and selective-failure blind.*

4.3 Security and Performance Comparison

We compared our scheme to other schemes [6,19,20] with hidden access structure in Table 1.

Table 1. Security and Performance Comparison

Scheme	Multi-Authority	Anonymity of Access Structure	Access Structure	Security Model	Ciphertext Size
LRZW's scheme [19]	No	Partially hidden	AND-gates on multi-valued attributes with wildcards	Selective-set	Linear
LOSTW's scheme [20]	No	Fully hidden	Inner product predicates	Fully secure	Linear
BK's scheme [6]	No	Fully hidden	AND, OR gates on multi-valued attributes	Selective-set	Linear
Our scheme	Yes	Fully hidden	AND, OR gates on multi-valued attributes	Selective-set	Linear

5 Conclusions

In this paper, we proposed a decentralized CP-ABE with fully hidden access structure. The access structure in our scheme is AND, OR gates on multi-valued attributes. Moreover, we considered user privacy from two aspects. On one hand, the access structure in our scheme is fully hidden, so intermediate user can get nothing about user attributes and policy from the access structure. On the other hand, malicious authorities cannot collaborate to collect user attributes by tracing user GID. The security of our scheme is proved under a standard DBDH complexity assumption.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (60842002, 61272542, 61103183, 61103184), the Fundamental Research Funds for the Central Universities(2013B07014, 2010B07114), the Six Talent Peaks Program of Jiangsu Province of China (2009182) and Program for New Century Excellent Talents in Hohai University.

References

1. Sahai, A., Waters, B.: Fuzzy Identity-Based Encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (2005)
2. 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. Proc. ACM Conf. Computer and Communications Security, pp. 89–98 (2006)
3. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-Policy Attribute-Based Encryption. In: SP 2007. IEEE Symposium on Security and Privacy, pp. 321–334 (2007)
4. 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. Proc. ACM Conf. Computer and Communications Security, pp. 195–203 (2007)
5. Cheung, L., Newport, C.: Provably Secure Ciphertext Policy ABE. In: Ning, P., di Vimercati, S.D.C., Syverson, P.F. (eds.) CCS 2007. Proc. ACM Conf. Computer and Comm. Security, pp. 456–465 (2007)
6. Balu, A., Kuppusamy, K.: Privacy Preserving Ciphertext Policy Attribute Based Encryption. In: Meghanathan, N., Boumerdassi, S., Chaki, N., Nagamalai, D. (eds.) CNSA 2010. CCIS, vol. 89, pp. 402–409. Springer, Heidelberg (2010)
7. Boneh, D., Waters, B.: Conjunctive, Subset, and Range Queries on Encrypted Data. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 535–554. Springer, Heidelberg (2007)
8. Nishide, T., Yoneyama, K., Ohta, K.: Attribute-Based Encryption with Partially Hidden Encryptor-Specified Access Structures. In: Bellovin, S.M., Gennaro, R., Keromytis, A.D., Yung, M. (eds.) ACNS 2008. LNCS, vol. 5037, pp. 111–129. Springer, Heidelberg (2008)
9. Lai, J., Deng, R.H., Li, Y.: Expressive CP-ABE with Partially Hidden Access Structures. In: Youm, H.Y., Won, Y. (eds.) ASIACCS 2012. Proc. ACM Conf. Computer and Communications Security, pp. 18–19 (2012)

10. Chase, M.: Multi-Authority Attribute Based Encryption. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 515–534. Springer, Heidelberg (2007)
11. Chase, M., Chow, S.S.M.: Improving Privacy and Security in Multi-Authority Attribute-Based Encryption. In: Al-Shaer, E., Jha, S., Keromytis, A.D. (eds.) CCS 2909. Proc. ACM Conf. Computer and Comm. Security, pp. 121–130 (2009)
12. Lewko, A., Waters, B.: Decentralizing Attribute-Based Encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 568–588. Springer, Heidelberg (2011)
13. Han, J., Susilo, W., Mu, Y., Yan, J.: Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption. IEEE Transantions on Parallel and Distributed System 23(11), 2150–2162 (2012), Nayak, A. (ed.)
14. Boneh, D., Boyen, X.: Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 223–238. Springer, Heidelberg (2004)
15. Beimel, A.: Secure Schemes for Secret Sharing and Key Distribution. PHD thesis, Israel Inst. of Technology, Technion, Haifa, Israel (1996)
16. Pedersen, T.P.: Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 129–140. Springer, Heidelberg (1992)
17. Camenisch, J., Stadler, M.: Efficient Group Signature Schemes for Large Groups. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 410–424. Springer, Heidelberg (1997)
18. 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)
19. Li, J., Ren, K., Zhu, B., Wan, Z.: Privacy-Aware Attribute-Based Encryption with User Accountability. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) ISC 2009. LNCS, vol. 5735, pp. 347–362. Springer, Heidelberg (2009)
20. Lewko, A.B., 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)