Anonymous Leakage-Resilient Ciphertext-Policy Attribute-Based Encryption Supporting Direct Revocation

Xiaoxu Gao(✉), Leyou Zhang, and Gongcheng Hu

School of Mathematics and Statistics, Xidian University, Xi'an 710126, China
gxx_xidian@163.com

Abstract. Leakage-resilient ciphertext-policy attribute-based encryption (LR-CP-ABE) is an important tool to achieve fine-grained access control of data and resist side-channel attacks. Privacy protection and user revocation are two practical problems it faces. However, most of the existing schemes fail to achieve user revocation while protecting user's privacy at present. To address the above problems, we propose an anonymous LR-CP-ABE scheme with user revocation in this paper, which is proven to be adaptively secure in the standard model under four static assumptions over composite order group. Furthermore, we also show the proposed scheme achieves the receivers anonymity which protects the users' privacy. The performance analyses confirm the feasibility of the proposed scheme.

Keywords: Anonymous · Ciphertext-policy attribute-based encryption · Direct revocation · Leakage-resilient

1 Introduction

ABE was first proposed by Sahai and Waters [1], which is an important tool for solving security and fine-grained data sharing and access control problems. It has become a research hotspot in recent years. The ABE systems can be divided into two categories: one is CP-ABE [2] and the other is key-policy ABE (KP-ABE) [3]. The most obvious difference between them is whether the private keys are related to the attribute set. In CP-ABE, a private key is associated with an attribute list, a ciphertext is related to an access structure. The users can decrypt the ciphertexts if and only if the user's attribute set satisfies the corresponding access structure. While in KP-ABE, the situation is reversed.

Revocation is a challenge problem in the CP-ABE setting because there has opportunities to dynamically change attributes or users. Therefore, the revocation mechanism can be divided into two types, namely, attribute revocation and user revocation. So far, there are two ways to solve this problem: direct

Supported by the National Cryptography Development Fund under grant (MMJJ20180209).

B. Shen et al. (Eds.): FCS 2019, CCIS 1105, pp. 21–35, 2019.
https://doi.org/10.1007/978-981-15-0818-9_2

revocation and indirect revocation. Indirect method means revocation mechanism by authority, which updates the private keys of a user who has not revoked an attribute periodically or dynamically. While direct method means revocation is performed by the date owner who specified the revoked user list during the encryption process. Although direct method has less flexibility in revoking users, it has an advantage in revoking costs. Revocable ABE was first proposed in [4,5], so far, it has made great progress, such as [6–9] and so on.

Although ABE can be directly applied to the design of secure access control, for the purpose of better protecting user's privacy and data security, anonymous ABE was proposed in [10,11] and further improved by [12,13]. More related works can refer to [14–19]. In anonymous ABE, the adversary cannot obtain some meaningful information about the corresponding attributes in the access policies.

However, studies have shown that these schemes can not resist various forms of attacks, such as side-channel attacks. Because the security of these schemes is based on an idealized assumption that the adversary cannot get any information of the private keys and internal state. In fact, this assumption is actually unrealistic. The adversary can learn meaningful information about the keys by using some of the physical information that the algorithm outputs. So the adversary can easily break the security of these schemes. In order to characterize the leaked information that the adversary available and protect the security of these schemes, ABE based on various leakage models are proposed in [20–27].

Zhang et al. [22] focused on the above three issues and designed a leakage-resilient secure ABE with fine-grained attribute revocation to achieve the semantic security in the continual key leakage model. Users need to pay a big price in decryption. Subsequently, Yu et al. [25] introduced a leakage-resilient CP-ABE supporting indirect revocation which can tolerate the leakage of the private keys and the master secret keys. The security of the scheme is proved by using dual system encryption.

While above schemes cannot achieve leakage-resilience, anonymity and user revocation at the same time. Therefore, it is worthwhile to study an efficient scheme that can realize the above three performances.

1.1 Our Contribution

In this paper, an CP-ABE scheme under the continuous leakage model is constructed whose leakage bound achieves $\lambda \leq (\omega - 1 - 2c) \log p_2$ during two updates, which is proved to be adaptively secure in the standard model under four static assumptions over composite order bilinear group. Moreover, this scheme can achieve the user's direct revocation by embedding the revocation list in the ciphertexts. We also give an analysis of how the scheme achieves anonymity (Table 1).

2 Preliminaries

2.1 Linear Secret Sharing

A secret sharing scheme Λ over a set of attributes S is called linear on the two conditions that:

Table 1. Symbols

Symbol	Description
Σ	A set of attributes. In other words, $\Sigma = \{att_1, att_2, ..., att_n\}$.
$p_{\check{i}}$	The orders of $\mathbb{G}_{p_{\check{i}}}$, where $\check{i} = 1, 2, 3, 4$
$g_{\check{i}}$	Generators of the subgroups $\mathbb{G}_{p_{\check{i}}}$ with order $p_{\check{i}}$, where $\check{i} = 1, 2, 3, 4$
$\mathbb{Z}_N$	The set of positive integers
pk	The public keys
msk	The master secret keys
$v_{i,j}$	The j^{th} value of att_i
sk_S	The private keys associated with attribute set $S = \{v_{1,x_1}, v_{2,x_2}, .., v_{n'',x_{n''}}\}$
m	Messages
CT	The ciphertexts
$x \in_R X$	Denote that x is randomly chosen from a set X
$\boldsymbol{A}$	A matrix
$\boldsymbol{v}$	A vector
$[n]$	A set of values from 1 to n
n_i	The possible values of the attribute att_i

(1) The shares for each attributes form a vector from $\mathbb{Z}_p$.
(2) There exists a $l \times n$ matrix $\boldsymbol{A}$ called sharing-generating matrix for Λ. The function ρ maps the x^{th} row of $\boldsymbol{A}$ to an attribute value labeling $\rho(x)$ for all $x \in [l]$. Then we selects a vector $\boldsymbol{v} = (s, v_2, ..., v_n) \in_R \mathbb{Z}_p^n$, where s is the secret to be shared, and $\boldsymbol{A} \cdot \boldsymbol{v}$ is the vector of l shares of the secret s according to Λ. The shares $(\boldsymbol{Av})_x$ belongs to the attribute value $\rho(x)$.

Linear Reconstruction. Let $C \in \Lambda$ be any authorized set, and let $I \subset \{1, 2, ..., l\}$ be defined as $I = \{x' | \rho(x') \in C\}$. Then, there exists constants $\{\mu_{x'} \in \mathbb{Z}_p\}_{x' \in I}$ such that, if $\{\lambda_{x'}\}$ are valid shares of any s in Λ, then $\sum_{x' \in I} \mu_{x'} \lambda_{x'} = s$. This collection $\{\mu_{x'}\}_{x' \in I}$ can be found in polynomial time.

2.2 Complexity Assumptions

Assumption 1. Given a instance $(\Theta = (N = p_1 p_2 p_3 p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_3, g_4, T)$, where $g_{\check{i}} \in_R \mathbb{G}_{p_{\check{i}}}$ for $\check{i} = 1, 3, 4$, the advantage of $\mathcal{A}$ distinguish $T \in_R \mathbb{G}_{p_1 p_4}$ from $T \in_R \mathbb{G}_{p_1 p_2 p_4}$ is negligible.

Assumption 2. Given instance $(\Theta = (N = p_1 p_2 p_3 p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_3, g_4, U_1 U_2, W_2 W_3, T)$, where $g_1, U_1 \in_R \mathbb{G}_{p_1}$, $U_2, W_2 \in_R \mathbb{G}_{p_2}$, $g_3, W_3 \in_R \mathbb{G}_{p_3}$ and $g_4 \in_R \mathbb{G}_{p_4}$, the advantage of $\mathcal{A}$ distinguish $T \in_R \mathbb{G}_{p_1 p_3}$ from $T \in_R \mathbb{G}_{p_1 p_2 p_3}$ is negligible.

Assumption 3. Given a instance $(\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_2, g_3, g_4, g_1^{\alpha}U_2, g_1^{s}W_2, g_2^{r}, U_2^{r}, T)$, where $s, \alpha, r \in_R \mathbb{Z}_N$, $g_1 \in_R \mathbb{G}_{p_1}$, $g_2, U_2, W_2 \in_R \mathbb{G}_{p_2}$ and $g_3 \in_R \mathbb{G}_{p_3}$, the advantage of $\mathcal{A}$ distinguish $T = \hat{e}(g,g)^{\alpha s}$ from $T \in_R \mathbb{G}_T$ is negligible.

Assumption 4. Given a instance $(\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_2, g_3, g_4, U_1U_4, U_1^{\hat{r}}U_2, g_1^{\hat{r}}W_2, g_1^{s}W_{24}, U_1g_3^{\hat{s}}, T)$, where $s, \hat{r}, \hat{s} \in_R \mathbb{Z}_N$, $g_1, U_1 \in_R \mathbb{G}_{p_1}$, $g_2, U_2, W_2 \in_R \mathbb{G}_{p_2}$, $g_3 \in_R \mathbb{G}_{p_3}$, $g_4, U_4 \in_R \mathbb{G}_{p_4}$ and $W_{24}, D_{24} \in_R \mathbb{G}_{p_2p_4}$, the advantage of $\mathcal{A}$ distinguish $T \in_R U_1^{s}D_{24}$ from $T \in_R \mathbb{G}_{p_1p_2p_4}$ is negligible.

2.3 Random Subspaces for Leakage Resilience over Arbitrary Functions

Theorem 1. *For any function $f : \mathbb{Z}_p^{m' \times d'} \to \phi$, there exists*

$$Dist((\boldsymbol{X}_1, f(\boldsymbol{X}_1\boldsymbol{T})), (\boldsymbol{X}_1, f(\boldsymbol{X}_2))) \leq \epsilon,$$

where $m', l', d' \in_R \mathbb{N}$, $2d' \leq l' \leq m'$, $\boldsymbol{X}_1 \in_R \mathbb{Z}_p^{m' \times l'}$, $\boldsymbol{X}_2 \in_R \mathbb{Z}_p^{m' \times d'}$, $\boldsymbol{T} \in_R Rank_{d'}(\mathbb{Z}_p^{l' \times d'})$, $|\phi| \leq 4(1 - \frac{1}{p}) \cdot p_2^{l'-2d'+1} \cdot \epsilon^2$.

Claim. For any function $f : \mathbb{Z}_p^{m'} \to \{0,1\}^{l'}$, there exists

$$Dist((\Delta, f(\boldsymbol{\mu})), (\Delta, f(\boldsymbol{\mu'}))) \leq \epsilon,$$

where $\Delta, \boldsymbol{\mu} \in_R \mathbb{Z}_p^{m'}$, $\boldsymbol{\mu'} \cdot \Delta = 0 \mod p, l' \leq 4p^{m'-3}(p-1) \cdot \epsilon^2$.

3 LR-CP-ABE Supporting Direct Revocation

3.1 Model of LR-CP-ABE with Direct Revocation

Three entities are included in our construction: attribute authority (AA), data owners (DO) and users.

Setup$(\kappa, \Sigma, \lambda)$: AA takes the security parameter κ, universe attribute set Σ and leakage bound λ as input, outputs the public keys pk and master secret keys msk.
KeyGen(pk, msk, S, id): AA inputs the public keys pk, master secret keys msk, attribute list S for the user with id, outputs the private keys sk_S.
UpdateUsk(pk, sk_S): AA takes the public keys pk and the secret keys sk_S as input, outputs the new private keys sk'_S.
Encrypt$(pk, m, \Lambda, \mathcal{R})$: DO takes the public keys pk, a message m, access structure Λ and revocation list $\mathcal{R}$ as input, then outputs the ciphertexts CT.
Decrypt(CT, sk_S): The users inputs the ciphertexts CT and the private keys sk_S, and outputs the message m.

3.2 Security Properties of the ANON-LR-CP-ABE with Direct Revocation

This game is played by the interaction between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, the concrete process is described as follows:

- **Setup**: $\mathcal{C}$ inputs the security parameter κ and the leakage upper bound λ, generates the public keys pk and the master secret keys msk. Then $\mathcal{C}$ sends pk to $\mathcal{A}$ while keeps msk. At the same time, $\mathcal{C}$ creates an initial empty lists: $\mathcal{L} = (hd, S, sk_S, L_{sk})$, where L_{sk} means the total leakage bits.
- **Phase 1**: $\mathcal{A}$ adaptively performs the following queries:
  - *KeyGen queries*: $\mathcal{A}$ sends an identity id and an attribute list S to $\mathcal{C}$, then $\mathcal{C}$ runs the algorithm **KeyGen** to generate the private keys sk_S. Finally, $\mathcal{C}$ updates $hd = hd + 1$ and adds the item $(hd, S, sk_S, 0)$ to the list $\mathcal{L}$.
  - *Leakage queries*: $\mathcal{A}$ gives a polynomial-time computable arbitrary function $f : \{0,1\}^* \to \{0,1\}^*$ to $\mathcal{C}$. Assume that the set is (hd, S, sk_S, L_{sk_S}), then $\mathcal{C}$ checks whether $|f(sk_S)| + L_{sk_S} \leq \lambda$. If this is true, it returns $f(sk_S)$ to $\mathcal{A}$. Otherwise, outputs the symbol $\perp$.
  - *UpdateUsk queries*: $\mathcal{A}$ queries the new updated secret keys for hd. If there is no (hd, S, sk_S, L_{sk_S}) found in set $\mathcal{L}$. Then $\mathcal{C}$ runs the algorithm **KeyGen** to get the private keys sk_S and sets $L_{sk_S} = 0$. Otherwise, $\mathcal{C}$ returns re-randomized private keys sk'_S with **UpdatedUsk** and updates the corresponding $L_{sk_S} = 0$.
- **Challenge**: $\mathcal{A}$ outputs two messages of the same length m_0, m_1, revocation list $\mathcal{R}$ and two challenge access structures $\Lambda_0(\boldsymbol{A}_0, \rho_0)$, $\Lambda_1(\boldsymbol{A}_1, \rho_1)$ to $\mathcal{C}$, then $\mathcal{C}$ selects $b \in \{0,1\}$ randomly and encrypts the message m_b under the access structure $\Lambda_b(\boldsymbol{A}_b, \rho_b)$. Finally, it outputs the ciphertexts CT^* to $\mathcal{A}$.
- **Phase 2**: The phase is similar to **Phase1** except that $\mathcal{A}$ cannot execute the *Leakage queries* and the *KeyGen queries* that the corresponding attribute set satisfies the challenge access structure.
- **Guess**: $\mathcal{A}$ outputs the guess b' of b and wins the game if $b' = b$.

If the advantage of $\mathcal{A}$ in the above game is negligible, then it is said that the anonymous CP-ABE scheme which supporting direct revocation is indistinguishable under the chosen plaintext attack (ANON-IND-CPA-REVO) and it is λ leakage-resilient, where the advantage of $\mathcal{A}$ is defined as

$$Adv_{\mathcal{A}}^{ANON-IND-CPA-REVO} = |\Pr[b' = b] - \frac{1}{2}|$$

4 Construction

4.1 Concrete Construction

Setup$(\kappa, \Sigma, \lambda)$**:** AA takes a security parameter κ and the attribute universe description Σ and a leakage bound λ as input. Then it runs the bilinear group generator to produce $\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e})$, defines $negl = p_2^{-c}$ as the

allowable maximum probability in succeeding in leakage guess and computes $\omega = \lceil 1 + 2c + \frac{\lambda}{\log p_2} \rceil$, where c is a positive constant. Then the algorithm generates the public keys as follows. First, it selects $g_1, h \in \mathbb{G}_{p_1}$, $g_3 \in \mathbb{G}_{p_3}$ and $a, \alpha \in \mathbb{Z}_N$ at random. Second, it selects $\boldsymbol{\rho} \in_R \mathbb{Z}_N^\omega$ and selects $t_{i,j} \in_R \mathbb{Z}_N$, $g_4, w_0, w_{i,j} \in_R \mathbb{G}_{p_4}$ for each $i \in [n], j \in [n_i]$, the public keys are

$$pk = \left(N, a_0, h, u, g_3, g_1^{\boldsymbol{\rho}}, y, T_{i,j}; \forall i \in [n], j \in [n_i] \right)$$

where $a_0 = g_1 w_0, u = g_1^a g_4, y = e(g_1, g_1)^\alpha, T_{i,j} = g_1^{t_{i,j}} w_{i,j}$.
The master secret keys are

$$msk = (a, \alpha, t_{i,j}, g_1).$$

KeyGen(pk, msk, S, id)**:** On input the public keys pk, the master keys msk, an attribute set S and users identity id, AA outputs the secret keys $sk_S = \left(S, sk_{S,1}, sk_{S,2} \right) = \left(S, \{\boldsymbol{k}_0, k_1\}, \{k_{2,i}, k_{3,i}, k_{4,i}\}_{i \in S} \right)$ and sends them to users. AA selects $r_{id}, y_1 \in_R \mathbb{Z}_N$, $\boldsymbol{y}_0, \boldsymbol{\sigma} \in_R \mathbb{Z}_N^\omega$ and picks $r_{i,j}, y_{i,j,2}$, $y_{i,j,3}, y_{i,j,4} \in_R \mathbb{Z}_N$ for $v_{i,j} \in S$, calculates and outputs the secret keys as follows.

$$\begin{aligned} sk_S &= \left(S, sk_{S,1}, sk_{S,2} \right) \\ &= \left(S, \{\boldsymbol{k}_0, k_1\}, \{k_{i,j,2}, k_{i,j,3}, k_{i,j,4}\}_{v_{i,j} \in S} \right) \\ &= \Big(S, \{ g_1^{\boldsymbol{\sigma}} * g_3^{\boldsymbol{y}_0}, g_1^{\alpha + a r_{id} + \langle \boldsymbol{\sigma}, \boldsymbol{\rho} \rangle} g_3^{y_1} \}, \{ g_1^{\alpha r_{id} + t_{i,j} r_{i,j} + a r_{i,j}} g_3^{y_{i,j,2}}, g_1^{r_{i,j}} g_3^{y_{i,j,3}}, \\ &\quad (g_1^{aid} h)^{r_{i,j}} g_3^{y_{i,j,4}} \}_{v_{i,j} \in S} \Big) \end{aligned} \tag{1}$$

UpdateSk(sk_S, S)**:** AA selects $\Delta r_{id}, \Delta y_1 \in_R \mathbb{Z}_N$, $\Delta \boldsymbol{\sigma}, \Delta \boldsymbol{y}_0 \in_R \mathbb{Z}_N^\omega$ and $\Delta r_{i,j}, \Delta y_{i,j,2}, \Delta y_{i,j,3}, \Delta y_{i,j,4} \in_R \mathbb{Z}_N$ for $v_{i,j} \in S$, outputs the re-randomized keys sk'_S:

$$\begin{aligned} sk'_S &= \left(S, sk'_{S,1}, sk'_{S,2} \right) \\ &= \left(S, \{\boldsymbol{k}'_0, k'_1\}, \{k'_{i,j,2}, k'_{i,j,3}, k_{i,j,4}\}_{v_{i,j} \in S} \right) \\ &= \Big(S, \{ \boldsymbol{k}_0 * g_1^{\Delta \boldsymbol{\sigma}} * g_3^{\Delta \boldsymbol{y}_0}, k_1 g_1^{a \Delta r_{id} + \langle \boldsymbol{\rho}, \Delta \boldsymbol{\sigma} \rangle} g_3^{\Delta y_1} \}, \{ k_{i,j,2} g_1^{\alpha \Delta r_{id} + t_{i,j} \Delta r_{i,j} + a \Delta r_{i,j}} \\ &\quad g_3^{\Delta y_{i,j,2}}, k_{i,j,3} g_1^{\Delta r_{i,j}} g_3^{\Delta y_{i,j,3}}, k_{i,j,4} (g_1^{aid} h)^{\Delta r_{i,j}} g_3^{\Delta y_{i,j,4}} \}_{v_{i,j} \in S} \Big) \end{aligned} \tag{2}$$

Encrypt(pk, m, Λ)**:** $\boldsymbol{A}$ in $\Lambda(\boldsymbol{A}, \rho)$ is a secret sharing matrix of $l \times n$, where ρ maps rows of $\boldsymbol{A}$ into attribute values. $\mathcal{R} = \{R_{\rho(x)}\}_{x \in [l]}$ be an attribute

revocation list. DO selects $\boldsymbol{v} = (s, v_2, ..., v_n) \in \mathbb{Z}_N^n$ at random. The revocation list of attribute $\rho(x)$ is $R_{\rho(x)} = \{id_1, id_2, ..., id_{l_x}\}$, where l_x is a variable number of revocation users. Then the algorithm selects $s_{x,i'} \in_R \mathbb{Z}_N$ for each $id_{i'} \in R_{\rho(x)}$ with the restriction that $\sum_{i'=1}^{l_x} s_{x,i'} = \lambda_x$ where $\lambda_x = \boldsymbol{A}_x \cdot \boldsymbol{v}$, $g_4, w_1, w_{\lambda_x,1}, w_{\lambda_x,2}, w_{x,i',1}, w_{x,i',2} \in_R \mathbb{G}_{p_4}$, $\boldsymbol{A}_x$ is the x^{th} row of $\boldsymbol{A}$. Finally, the algorithm outputs the ciphertexts CT as follows:

$$\begin{aligned} CT &= \left(\boldsymbol{A}, \{I_x\}_{x\in[l]}, \mathcal{R}, c_0, \boldsymbol{c}_1, c_2, \{c_{x,0}, c_{x,1}, \{c_{x,i'}^1, c_{x,i'}^2\}_{i'\in\{1,2,...,l_x\}}\}_{x\in[l]}\right) \\ &= \Bigg(\boldsymbol{A}, \{I_x\}_{x\in[l]}, \mathcal{R}, my^s, a_0^{-s\boldsymbol{\rho}} * g_4^{\boldsymbol{\mu}}, a_0^s \cdot w_1, \{a_0^{\lambda_x} \cdot w_{\lambda_x,1}, T_{\rho(x)}^{\lambda_x} \cdot w_{\lambda_x,2}, \\ &\quad \{a_0^{s_{x,i'}} \cdot w_{x,i',1}, (u^{id_{i'}}h)^{s_{x,i'}} \cdot w_{x,i',2}\}_{i'\in\{1,2,...,l_x\}}\}_{x\in[l]}\Bigg) \end{aligned} \tag{3}$$

where $\boldsymbol{\mu} \in_R \mathbb{Z}_N^{\omega}$, $\{I_x\}_{x\in[l]} \subset \{1, 2, ..., n\}$ is the index set of corresponding attribute name.

Decrypt(CT, sk_S)**:** This algorithm takes the public keys pk, users identity id, the ciphertexts CT and the secret keys sk_S as input. If $id \in R_{\rho(x)}$, then the algorithm aborts. Otherwise, suppose $\mathcal{H} = \{x|\rho(x) \in S, id \notin R_{\rho(x)}\}$. If $S' = \{\rho(x)|x \in \mathcal{H}\}$ satisfies the access structure, then users computes $d_{x,1}, d_{x,2}$ for every $x \in \mathcal{H}$ at first.

$$\begin{aligned} d_{x,1} &= \prod_{i'=1}^{l_x} \left(\frac{\hat{e}(k_{\rho(x),3}, c_{x,i'}^2)}{\hat{e}(k_{\rho(x),4}, c_{x,i'}^1)}\right)^{\frac{1}{id-id_{i'}}} \\ &= \prod_{i'=1}^{l_x} \left(\frac{\hat{e}(g_1^{r_{\rho(x)}} g_3^{y_{\rho(x),3}}, (u^{id_{i'}}h)^{s_{x,i'}})}{\hat{e}((g^{aid}h)^{r_{\rho(x)}} g_3^{y_{\rho(x),4}}, g_1^{s_{x,i'}})}\right)^{\frac{1}{id-id_{i'}}} \\ &= \prod_{i'=1}^{l_x} \hat{e}(g_1, g_1)^{-ar_{\rho(x)}s_{x,i'}} \\ &= \hat{e}(g_1, g_1)^{-a\lambda_x r_{\rho(x)}} \end{aligned} \tag{4}$$

$$\begin{aligned} d_{x,2} &= \frac{\hat{e}(k_{\rho(x),2}, c_{x,0})}{\hat{e}(k_{\rho(x),3}, c_{x,1})} \\ &= \frac{\hat{e}(g_1^{\alpha r_{id}+t_{\rho(x)}r_{\rho(x)}+ar_{\rho(x)}} g_3^{y_{\rho(x),2}}, g^{\lambda_x})}{\hat{e}(g_1^{r_{\rho(x)}} g_3^{y_{\rho(x),3}}, T_{\rho(x)}^{\lambda_x})} \\ &= \hat{e}(g_1, g_1)^{\alpha r_{id}\lambda_x + a\lambda_x r_{\rho(x)}} \end{aligned} \tag{5}$$

Obvious, there are

$$\begin{aligned} d_x &= d_{x,1}d_{x,2} = \hat{e}(g_1, g_1)^{\alpha r_{id}\lambda_x} \\ CT' &= \prod_{x\in\mathcal{H}} d_x^{\mu_x} = \hat{e}(g_1, g_1)^{\alpha r_{id}s} \end{aligned} \tag{6}$$

where $\sum_{x\in\mathcal{H}} \mu_x \boldsymbol{A}_x = (1, 0, 0, .., 0)$. Finally, it computes the $m = \frac{c_0}{\hat{e_\omega}(\boldsymbol{c}_1, \boldsymbol{k}_0)\hat{e}(c_2, k_1)} CT'$.

4.2 Security Proof

The security proof is based on dual system encryption, so we define the semi-functional keys and semi-functional ciphertexts as follows:

Semi-functional Keys: There are two types of semi-functional keys in our proof. Firstly, we run the **KeyGen** to get normal private keys as: $sk_S = \left(S, sk_{S,1}, sk_{S,2}\right) = \left(S, \{\boldsymbol{k}'_0, k'_1\}, \{k'_{i,j,2}, k'_{i,j,3}, k_{i,j,4}\}_{v_{i,j} \in S}\right)$. Then it selects $\boldsymbol{d}_0 \in_R \mathbb{Z}_N^\omega, d_1 \in_R \mathbb{Z}_N$ and $d_{i,j,2}, d_{i,j,3}, d_{i,j,4} \in_R \mathbb{Z}_N$ for $v_{i,j} \in S$ and compute two types of semi-functional private keys components as follows.

Type 1.

$$\boldsymbol{k}_0 = \boldsymbol{k}'_0 * g_2^{\boldsymbol{d}_0}, \quad k_1 = k'_1 g_2^{d_1}, \quad k_{i,j,2} = k'_{i,j,2} g_2^{d_{i,j,2}},$$
$$k_{i,j,3} = k'_{i,j,3} g_2^{d_{i,j,3}}, \quad k_{i,j,4} = k'_{i,j,4} g_2^{d_{i,j,4}}.$$

Type 2.

$$\boldsymbol{k}_0 = \boldsymbol{k}'_0 * g_2^{\boldsymbol{d}_0}, \quad k_1 = k'_1 g_2^{d_1}, \quad k_{i,j,2} = k'_{i,j,2} g_2^{d_{i,j,2}},$$
$$k_{i,j,3} = k'_{i,j,3}, \quad k_{i,j,4} = k'_{i,j,4}.$$

Semi-functional Ciphertexts: For an access structure $\Lambda(\boldsymbol{A}, \rho)$ and a revocation list $\mathcal{R}$, we first run the encryption algorithm **Encrypt** to obtain normal ciphertexts $CT = \left(\boldsymbol{A}, \{I_x\}_{x \in [l]}, \mathcal{R}, c_0, \boldsymbol{c}'_1, c'_2, \{c'_{x,0}, c'_{x,1}, \{c^{1'}_{x,i'}, c^{2'}_{x,i'}\}_{i' \in \{1,2,\ldots,l_x\}}\}_{x \in [l]}\right)$ and choose some random elements $\boldsymbol{e}_1 \in \mathbb{Z}_N^\omega$ and $e_2, e_{x,0}, e_{x,1}, e_{x,i',1}, e_{x,i',2} \in \mathbb{Z}_N$. The semi-functional ciphertexts are computed as follows:

$$\boldsymbol{c}_1 = \boldsymbol{c}'_1 * g_2^{\boldsymbol{e}_1}, \quad c_2 = c'_2 g_2^{e_2}, \quad c_{x,0} = c'_{x,0} g_2^{e_{x,0}},$$
$$c_{x,1} = c'_{x,1} g_2^{e_{x,1}}, \quad c^1_{x,i'} = c^{1'}_{x,i'} g_2^{e_{x,i',1}}, \quad c^2_{x,i'} = c^{2'}_{x,i'} g_2^{e_{x,i',2}}.$$

The security of the program is proved by a series of indistinguishable games. The specific game definitions are described below:

$Game_{real}$: This is a real game that the private keys and ciphertexts are in normal form.
$Game_0$: The game is similar to the $Game_{real}$ except that the ciphertexts are semi-functional.
$Game_{k-1,2}$: The first $k-1$ private keys are semi-functional of **Type 2**, the rest of private keys are normal.
$Game_{k,1}$: The game is similar to the $Game_{k-1,2}$ except the k^{th} private key is semi-functional of **Type 1**.
$Game_{k,2}$: The game is similar to the $Game_{k,1}$ except the k^{th} private key is semi-functional of **Type 2**.

$Game_{q,2}$: All of private keys are semi-functional of **Type 2** and the ciphertexts are semi-functional, where q is the number of queries.
$Game_{final,0}$: The ciphertext component c_0 is the encryption of a random message.
$Game_{final,1}$: The component $c_{x,i',2}$ is a random element in subgroup $\mathbb{G}_{p_1p_2p_4}$.

Lemma 1. *Suppose that there is an adversary $\mathcal{A}$ can distinguish the $Game_{real}$ and $Game_0$ with a non-negligible advantage ϵ, then there is a simulator $\mathcal{B}$ breaks the Assumption 1 with same advantage.*

Proof. $\mathcal{B}$ receives the challenge instance $(\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_3, g_4, T)$ from the challenge $\mathcal{C}$ and simulates the $Game_{real}$ or $Game_0$.

Setup: After receiving the challenge instance $(\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_3, g_4, T)$, $\mathcal{B}$ generates the public keys as follows: $pk = \Big(N, a_0 = g_1 g_4^{a'}, h = g_1^t, u = g_1^a g_4, g_3, g_1^{\boldsymbol{\rho}}, y = e(g_1, g_1)^{\alpha}, T_{i,j} = g_1^{t_{i,j}} g_4^{a_{i,j}}; \forall i \in [n], j \in [n_i]\Big)$, where $t, a, a', \alpha, t_{i,j}, a_{i,j} \in_R \mathbb{Z}_N, \boldsymbol{\rho} \in_R \mathbb{Z}_N^{\omega}$.

Phase 1: Because $\mathcal{B}$ knows the master keys, so it can answer all the *KeyGen queries* and *Leakage queries*.

Challenge: $\mathcal{A}$ sends two challenge access structure $\Lambda_0^*(\boldsymbol{A}_0^*, \rho_0^*), \Lambda_1^*(\boldsymbol{A}_1^*, \rho_1^*)$, two messages m_0, m_1 of equal length and a revocation list $\mathcal{R} = \{R_{\rho(x)}\}_{x\in[l]}$ to $\mathcal{B}$, then $\mathcal{B}$ selects $b \in \{0,1\}$ at random and computes the ciphertexts as follows: $CT = \Big(\boldsymbol{A}_b^*, \{I_{b,x}\}_{x\in[l]}, \mathcal{R}, c_0 = m_b \hat{e}(T, g)^{\alpha}, \boldsymbol{c}_1' = T^{-\boldsymbol{\rho}} g_4^{\boldsymbol{u}}, c_2' = T g_4^{w_1'}, \{c_{x,0}' = T^{\lambda_{b,x}} g_4^{w'_{\lambda_{b,x},1}}, c_{x,1}' = T^{t_{\rho(x)}\lambda_{b,x}} g_4^{w'_{\lambda_{b,x},2}}, \{c_{x,i'}^{1'} = T^{s_{x,i'}} g_4^{w'_{x,i',1}}, c_{x,i'}^{2'} = T^{(aid'+t)s_{x,i'}} g_4^{w'_{x,i',2}}\}_{i'\in\{1,2,\ldots,l_x\}}\}_{x\in[l]}\Big)$, where $\lambda_{b,x} = \boldsymbol{A}_{b,x}^* \cdot \boldsymbol{v}', \boldsymbol{u}, \boldsymbol{v}' = (1, v_2', v_3', ..., v_n') \in_R \mathbb{Z}_N^{\omega}$, $\sum_{i'=1}^{l_x} s_{x,i'} = \lambda_{b,x}$, $w_1', w'_{\lambda_{b,x},1}, w'_{\lambda_{b,x},2}, w'_{x,i',1}, w'_{x,i',2}, s_{x,i'} \in_R \mathbb{Z}_N$, $\{I_x\}_{x\in[l]}$ is the index set of corresponding attribute name.

Phase 2: Same as **Phase 1** except that $\mathcal{A}$ cannot execute the *Leakage queries* and *KeyGen queries* that the corresponding attribute set satisfies the challenge access structure.

Guess: $\mathcal{A}$ outputs the guess of b' of b. If $b' = b$, $\mathcal{A}$ wins the game.

If $T \in_R \mathbb{G}_{p_1p_4}$, then $\mathcal{B}$ simulates the $Game_{real}$. Otherwise, $\mathcal{B}$ simulates the $Game_0$. Therefore, if $\mathcal{A}$ can distinguish these two games with a non-negligible advantage, then $\mathcal{B}$ can break the Assumption 1 with same advantage.

Lemma 2. *Suppose that there is an adversary $\mathcal{A}$ can distinguish the $Game_{k-1,2}$ and $Game_{k,1}$ with a non-negligible advantage ϵ, then there is a simulator $\mathcal{B}$ breaks the Assumption 2 with same advantage.*

Proof. $\mathcal{B}$ receives the challenge instance $(\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_3, g_4, U_1U_2, W_2W_3, T)$ from the challenge $\mathcal{C}$ and simulates the $Game_{k-1,2}$ or $Game_{k,1}$.

Setup: The algorithm of **Setup** is same as that in Lemma 1.

KeyGen queries in **Phase 1**: To generate the first $k-1$ semi-functional keys, $\mathcal{B}$ chooses $r_{id}, y_1 \in \mathbb{Z}_N$ at random, $\boldsymbol{y}_0, \boldsymbol{\sigma} \in_R \mathbb{Z}_N^{\omega}$ and $r_{i,j}, y_{i,j,2}, y_{i,j,3}, y_{i,j,4} \in_R \mathbb{Z}_N$ for $v_{i,j} \in S$, calculates and outputs the secret keys of **Type 1** as follows.

$$
\begin{aligned}
sk_S &= \Big(S, sk_{S,1}, sk_{S,2}\Big) \\
&= \Big(S, \{\boldsymbol{k}_0, k_1\}, \{k_{i,j,2}, k_{i,j,3}, k_{i,j,4}\}_{v_{i,j}\in S}\Big) \\
&= \Big(S, \{g_1^{\boldsymbol{\sigma}} * (W_2W_3)^{\boldsymbol{y}_0}, g_1^{\alpha + ar_{id} + \langle \boldsymbol{\sigma}, \boldsymbol{\rho}\rangle}(W_2W_3)^{y_1}\}, \{g_1^{\alpha r_{id} + t_{i,j} r_{i,j} + ar_{i,j}} \\
&(W_2W_3)^{y_{i,j,2}}, g_1^{r_{i,j}} g_3^{y_{i,j,3}}, (g_1^{aid}h)^{r_{i,j}} g_3^{y_{i,j,4}}\}_{v_{i,j}\in S}\Big)
\end{aligned}
$$

To generate the k^{th} private key, $\mathcal{B}$ picks $r_{id}, y_1 \in \mathbb{Z}_N$ randomly, $\boldsymbol{y}_0, \boldsymbol{\sigma'} \in_R \mathbb{Z}_N^{\omega}$ and $r_{i,j}, y_{i,j,2}, y_{i,j,3}, y_{i,j,4} \in_R \mathbb{Z}_N$ for $v_{i,j} \in S$, outputs the following secret keys .

$$
\begin{aligned}
sk_S &= \Big(S, sk_{S,1}, sk_{S,2}\Big) \\
&= \Big(S, \{\boldsymbol{k}_0, k_1\}, \{k_{i,j,2}, k_{i,j,3}, k_{i,j,4}\}_{v_{i,j}\in S}\Big) \\
&= \Big(S, \{T^{\boldsymbol{\sigma}'} * g_3^{\boldsymbol{y}_0}, g_1^{\alpha} T^{a + \langle \boldsymbol{\sigma}', \boldsymbol{\rho}\rangle} g_3^{y_1}\}, \{T^{\alpha} g_1^{t_{i,j} r_{i,j} + ar_{i,j}} g_3^{y_{i,j,2}}, \\
&g_1^{r_{i,j}} g_3^{y_{i,j,3}}, (g_1^{aid}h)^{r_{i,j}} g_3^{y_{i,j,4}}\}_{v_{i,j}\in S}\Big)
\end{aligned}
$$

The rest of private keys are normal keys.

Challenge: $\mathcal{A}$ sends two challenge access structure $\Lambda_0^*(\boldsymbol{A}_0^*, \rho_0^*), \Lambda_1^*(\boldsymbol{A}_1^*, \rho_1^*)$, two message m_0, m_1 of equal length and a revocation list $\mathcal{R} = \{R_{\rho(x)}\}_{x\in[l]}$ to $\mathcal{B}$, then $\mathcal{B}$ selects $b \in \{0,1\}$ at random and calculates the ciphertexts as follows: $CT = \Big(\boldsymbol{A}_b^*, \{I_{b,x}\}_{x\in[l]}, \mathcal{R}, c_0 = m_b \hat{e}(U_1U_2, g)^{\alpha}, \boldsymbol{c}_1' = (U_1U_2)^{-\boldsymbol{\rho}} g_4^{\boldsymbol{u}}, c_2' = (U_1U_2) g_4^{w_1'}, \{c_{x,0}' = (U_1U_2)^{\lambda_{b,x}} g_4^{w'_{\lambda_{b,x},1}}, c_{x,1}' = (U_1U_2)^{t_{\rho(x)}\lambda_{b,x}} g_4^{w'_{\lambda_{b,x},2}}, \{c_{x,i'}^{1'} = (U_1U_2)^{s_{x,i}} g_4^{w'_{x,i',1}}, c_{x,i'}^{2'} = ((U_1U_2)^{(aid_{i'}+t)s_{x,i'}} g_4^{w'_{x,i',2}}\}_{i'\in\{1,2,\ldots,l_x\}}\}_{x\in[l]}\Big)$, where $\lambda_{b,x} = \boldsymbol{A}_x^* \cdot \boldsymbol{v}', \boldsymbol{u}, \boldsymbol{v}' = (1, v_2', v_3', ..., v_n') \in_R \mathbb{Z}_N^{\omega}$, $\sum_{i'=1}^{l_x} s_{x,i'} = \lambda_{b,x}$, $w_1', w'_{\lambda_{b,x},1}, w'_{\lambda_{b,x},2}, w'_{x,i',1}, w'_{x,i',2}, s_{x,i'} \in_R \mathbb{Z}_N$, $\{I_x\}_{x\in[l]}$ is the index set of corresponding attribute name.

Phase 2: Same as **Phase 2** in Lemma 1.

Guess: $\mathcal{A}$ outputs the guess of b' of b. If $b' = b$, $\mathcal{A}$ wins the game.

It can be learn from the analysis above that $\mathcal{B}$ simulates the $Game_{k-1,2}$ if $T \in_R \mathbb{G}_{p_1p_3}$. Vice versa. So if $\mathcal{A}$ distinguish these two games with a non-negligible advantage ϵ, then there is a simulator $\mathcal{B}$ break the Assumption 2 with same advantage.

Lemma 3. *Suppose that there is an adversary $\mathcal{A}$ can distinguish the $Game_{k,1}$ and $Game_{k,2}$ with a non-negligible advantage ϵ, then there is a simulator $\mathcal{B}$ breaks the Assumption 2 with same advantage.*

Proof. $\mathcal{B}$ receives the challenge instance $(\Theta = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e}), g_1, g_3, g_4, U_1U_2, W_2W_3, T)$ from the challenge $\mathcal{C}$ and simulates the $Game_{k,1}$ or $Game_{k,2}$.

The proof of Lemma 3 is similar to that of Lemma 2 except the construction of k^{th} private key.

$$
\begin{aligned}
sk_S &= \left(S, sk_{S,1}, sk_{S,2}\right) \\
&= \left(S, \{\boldsymbol{k}_0, k_1\}, \{k_{i,j,2}, k_{i,j,3}, k_{i,j,4}\}_{v_{i,j} \in S}\right) \\
&= \Big(S, \{g_1^{\boldsymbol{\sigma}'} * (W_2W_3)^{\boldsymbol{y}_0}, g_1^{\alpha+a+\langle \boldsymbol{\sigma}', \boldsymbol{\rho} \rangle} (W_2W_3)^{y_1}\}, \{g_1^{\alpha r_{id}} T^{t_{i,j}+a} (W_2W_3)^{y_{i,j,2}}, \\
&\quad T g_3^{y_{i,j,3}}, T^{aid+t} g_3^{y_{i,j,4}}\}_{v_{i,j} \in S}\Big)
\end{aligned}
$$

If $T \in_R \mathbb{G}_{p_1p_2p_3}$, then $\mathcal{B}$ simulates the $Game_{k,1}$. Otherwise, $\mathcal{B}$ simulates the $Game_{k,2}$. So if $\mathcal{A}$ can distinguish these two schemes with a non-negligible advantage, then there is a simulator $\mathcal{B}$ breaks the Assumption 2 with same advantage.

Lemma 4. *Suppose that there is an adversary $\mathcal{A}$ can distinguish the $Game_{q,2}$ and $Game_{final,0}$ with a non-negligible advantage ϵ, then there is a simulator $\mathcal{B}$ breaks the Assumption 3 with same advantage.*

Lemma 5. *Suppose that there is an adversary $\mathcal{A}$ distinguish the $Game_{final,0}$ and $Game_{final,1}$ with a non-negligible advantage ϵ, then there is a simulator $\mathcal{B}$ breaks the Assumption 4 with same advantage.*

We omitted the proof of Lemmas 4 and 5 because the space limitation.

Theorem 2. *If the Assumptions 1, 2, 3 and 4 hold, then our scheme is λ-leakage-resilient and anonymous for $\lambda \leq (\omega - 1 - 2c) \log p_2$, where c is a fixed positive constant.*

Proof. If these four assumptions hold, then from the Lemmas 1, 2, 3, 4 and 5, our scheme is λ-leakage-resilient and anonymous for $\lambda \leq (\omega - 1 - 2c) \log p_2$, where c is a fixed positive constant.

4.3 Leakage Performance

In this part, we give a concrete analysis of leakage resilience. The scheme has the same leakage bound $\lambda \leq (\omega - 1 - 2c)\log p_2$ with schemes [20–22] and the allowable probability $negl = p_2^{-c}$. Thus, the leakage rate of our scheme is $\gamma = \frac{\omega-1-2c}{(1+c_1+c_3)(\omega+1+3|S|)}$, where $p_{\mathbb{i}}(\mathbb{i} \in [4])$ is large primes of $d_{\mathbb{i}} = c_{\mathbb{i}}\kappa$ bits respectively. $c_{\mathbb{i}}$ is a positive constant.

4.4 Anonymity Analysis

To achieve the anonymity, we add the random elements in $\mathbb{G}_{p_4}$ to components of public keys and the ciphertexts which has no effect on the decryption process because orthogonality. Next, we will give a concrete process to explain how to achieve anonymity.

$$\begin{aligned} \hat{e}(c_{x,1}, a_0) &= \hat{e}(T_{\rho(x)}^{\lambda_x} \cdot w_{\lambda_x,2}, g_1 \cdot w_0) \\ &= \hat{e}(g_1, g_1)^{t_{\rho(x)}\lambda_x} \hat{e}(w_{\rho(x)} w_{\lambda_x,2}, w_0)^{\lambda_x} \end{aligned} \tag{7}$$

$$\begin{aligned} \hat{e}(c_{x,0}, T_{i,j}) &= \hat{e}(a_0^{\lambda_x} \cdot w_{\lambda_x,1}, g_1^{t_{i,j}} \cdot w_{i,j}) \\ &= \hat{e}(g_1, g_1)^{t_{i,j}\lambda_x} \hat{e}(w_0 w_{\lambda_x,1}, w_{i,j})^{\lambda_x} \end{aligned} \tag{8}$$

In this case, we cannot decide the attribute value $\rho(x)$ in the access policy from the DDH-test even if $v_{i,j} = \rho(x)$, where $v_{i,j}$ is the attribute value for testing.

5 Performance Analysis

In this section, we will give a detailed analysis of the different schemes in terms of performance and efficiency in Tables 2 and 3, respectively.

As shown in Table 2, we compare these schemes [9,22,25,27] with our construction in terms of revoidability, leakage-resilient and anonymity. [9,22,25] can support revocation, but all of them are not anonymous. In addition, [9] is not leakage-resilient. [27] cannot support revocation. However, our construction can achieve these three goals simultaneously.

Table 2. Performance comparisons among different ABE schemes

2 Scheme	Support revocation	Leakage-resilient	Anonymous
[9]	✓	×	×
[22]	✓	✓	×
[25]	✓	✓	×
[27]	×	✓	✓
Ours	✓	✓	✓

Let $\|\mathbb{G}\|, \|\mathbb{G}_T\|$ represent the size of the group $\mathbb{G}$ and $\mathbb{G}_T$ respectively. n is the number of attributes in universe attribute set, $|S|$ is the number of attributes in an attribute list S, l is the number of rows in $\boldsymbol{A}$, n' is the maximum number of users in the system. ω is the leakage parameter and P is the time of pairing operation.

Table 3. Efficiency comparisons among different ABE schemes

Scheme	Public parameter size	Private key size	Ciphertext size	Decryption time
[9]	$(2n+2)\|\mathbb{G}\|$	$(2+\|S\|)\|\mathbb{G}\|$	$(1+2l)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(1+2\|S\|)P$
[22]	$(\omega+n+2n')$ $\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+2\|S\|)\|\mathbb{G}\|$	$(\omega+5l)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+4)\|\mathcal{R}\|P$
[25]	$(\omega+3+n)$ $\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+2+\|S\|)\|\mathbb{G}\|$	$(\omega+1+$ $2l)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+1+2\|S\|)P$
[27]	$(\omega+3+n)$ $\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+2+\|S\|)\|\mathbb{G}\|$	$(\omega+1+$ $2l)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+1+2\|S\|)P$
Ours	$(\omega+4+nn_i)$ $\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\omega+1+3\|S\|)\|\mathbb{G}\|$	$\|\mathbb{G}_T\|+(\omega+1+l+$ $\sum_{x=1}^{l} l_x)\|\mathbb{G}\|$	$(\omega+1+\sum_{x=1}^{\|\mathcal{H}\|}(2l_x+2))P$

6 Conclusions

In this paper, a leakage-resilient CP-ABE scheme is proposed, which supports direct revocation and achieves adaptive security under four static assumptions in the standard model. Additionally, we show the proposed scheme achieves the anonymity based on the dual system encryption and composite order group. The performance analyses confirm the feasibility of our scheme. However, the proposed scheme relies on the composite order group, which issues a higher computation cost than a scheme in a prime order group under the same security standard. Designing a scheme with the same properties which is based on prime order bilinear group will be our future work.

References

1. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 457–473. ACM, Aarhus (2005)
2. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: IEEE Symposium on Security and Privacy, pp. 321–334. IEEE, Berkeley (2007)
3. Goyal, V., Pandey, O., Sahai, A., et al.: Attribute-based encryption for fine grained access control of encrypted data. In: 13th ACM Conference on Computer and Communications Security, pp. 89–98. ACM, Alexandria (2006)
4. Yu, S., Wang, C., Ren, K., et al.: Attribute based data sharing with attribute revocation. In: 5th ACM Symposium on Information, Computer and Communications Security, pp. 261–270. ACM, Beijing (2010)

5. Liang, X., Lu, R., Lin, X., et al.: Ciphertext policy attribute based encryption with efficient revocation (2010)
6. Hur, J., Noh, D.K.: Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Trans. Parallel Distrib. Syst. **22**(7), 1214–1221 (2011)
7. Zhang, Y., Chen, X., Li, J., et al.: FDR-ABE: attribute-Based encryption with flexible and direct revocation. In: 5th International Conference on Intelligent Networking and Collaborative Systems, pp. 38–45. IEEE, Xi'an (2013)
8. Xie, X., Ma, H., Li, J., et al.: An efficient ciphertext-policy attribute-based access control towards revocation in cloud computing. J. UCS **19**(16), 2349–2367 (2013)
9. Naruse, T., Mohri, M., Shiraishi, Y.: Provably secure attribute-based encryption with attribute revocation and grant function using proxy re-encryption and attribute key for updating. Hum.-Centric Comput. Inf. Sci. **5**(1), 1–13 (2015)
10. Kapadia, A., Tsang, P., Smith, S.M.: Attribute-based publishing with hidden credentials and hidden policies. NDSS **7**, 179–192 (2007)
11. Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: 27th Annual International Conference on Advances in Cryptology, pp. 146–162. ACM, Istanbul (2008)
12. Nishide, T., Yoneyama, K., Ohta, K.: Attribute-based encryption with partially hidden encryptor-specified access structures. In: 6th International Conference on Applied Cryptography and Network Security, pp. 111–129. ACM, New York (2008)
13. Li, J., Ren, K., Zhu, B., et al.: Privacy-aware attribute-based encryption with user accountability. In: 12th International Conference on Information Security, pp. 347–362. ACM, Pisa (2009)
14. Han, F., Qin, J., Zhao, H., et al.: A general transformation from KP-ABE to searchable encryption. Future Gener. Comput. Syst. **30**(1), 107–115 (2014)
15. Zhang, Y., Chen, X., Li, J., et al.: Anonymous attribute-based encryption supporting efficient decryption test. In: 8th ACM SIGSAC symposium on Information, Computer and Communications Security, pp. 511–516. ACM, Hangzhou (2013)
16. Chaudhari, P., Das, M.L., Mathuria, A.: On anonymous attribute based encryption. In: Jajodia, S., Mazumdar, C. (eds.) ICISS 2015. LNCS, vol. 9478, pp. 378–392. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26961-0_23
17. Zhang, Y., Zheng, D.: Anonymous attribute-based encryption with large universe and threshold access structures. In: IEEE International Conference on Computational Science and Engineering and IEEE International Conference on Embedded and Ubiquitous Computing, pp. 870–874. IEEE, Guangzhou (2017)
18. Zhang, L., Cui, Y., Mu, Y.: Improving privacy-preserving CP-ABE with hidden access policy. In: Sun, X., Pan, Z., Bertino, E. (eds.) ICCCS 2018. LNCS, vol. 11065, pp. 596–605. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00012-7_54
19. Zhang, L., Hu, G., Mu, Y., et al.: Hidden ciphertext policy attribute-based encryption with fast decryption for personal health record system. IEEE Access **7**, 33202–33213 (2019)
20. Lewko, A., Rouselakis, Y., Waters, B.: Achieving leakage resilience through dual system encryption. In: 8th Conference on Theory of Cryptography, pp. 70–88. ACM (2011)
21. Zhang, M., Shi, W., Wang, C., Chen, Z., Mu, Y.: Leakage-resilient attribute-based encryption with fast decryption: models, analysis and constructions. In: Deng, R.H., Feng, T. (eds.) ISPEC 2013. LNCS, vol. 7863, pp. 75–90. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38033-4_6

22. Zhang, M.: New model and construction of ABE: achieving key resilient-leakage and attribute direct-revocation. In: Susilo, W., Mu, Y. (eds.) ACISP 2014. LNCS, vol. 8544, pp. 192–208. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08344-5_13
23. Wang, Z., Yiu, S.M.: Attribute-based encryption resilient to auxiliary input. In: Au, M.-H., Miyaji, A. (eds.) ProvSec 2015. LNCS, vol. 9451, pp. 371–390. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26059-4_21
24. Zhang, L., Zhang, J., Hu, Y.: Attribute-based encryption resilient to continual auxiliary leakage with constant size ciphertexts. J. China Univ. Posts Telecommun. **23**(3), 18–28 (2016)
25. Yu, Q., Li, J.: Continuous leakage resilient ciphertext-policy attribute-based encryption supporting attribute revocation. Comput. Eng. Appl. **52**(20), 29–38 (2016)
26. Zhang, L., Zhang, J., Mu, Y.: Novel leakage-resilient attribute-based encryption from hash proof system. Comput. J. **60**(4), 541–554 (2017)
27. Zhang, J., Zhang, L.: Anonymous CP-ABE against side-channel attacks in cloud computing. J. Inf. Sci. Eng. **33**(3), 789–805 (2017)