Efficient Anonymous Ciphertext-Policy Attribute-based Encryption for General Structures Supporting Leakage-Resilience

Xiaoxu Gao and Leyou Zhang
(Corresponding author: Xiaoxu Gao)
School of Mathematics and Statistics, Xidian University
Xi'an, Shaanxi 710071, China
(Email: gxx_xidian@163.com)
(Received Mar. 5, 2019; Revised and Accepted Nov. 6, 2019; First Online Jan. 23, 2020)

Abstract

The traditional cryptographic schemes cannot guarantee data security and users privacy under the side-channel attacks. Additionally, most of the existing leakage-resilient schemes cannot protect the privacy of the receivers. To achieve the leakage resilience and privacy-preserving, two anonymous attribute-based encryption (ABE) schemes for general access structures are proposed. In the first scheme, the access structure is encoded as minimal sets which provide the higher efficiency in the cost of decryption algorithm. Then we show how to obtain an anonymous leakage-resilient ABE for non-monotone access structures. Both schemes can tolerate the continual leakage when an update algorithm is employed in the event of the occurrence of the leakage information beyond the allowable leakage bound. They are proven to be adaptively secure in the standard model under four static assumptions over composite order bilinear group. The performance analyses confirm efficiency of our schemes.

Keywords: Anonymous; Attribute-based Encryption; Ciphertext-Policy; Leakage-Resilience

1 Introduction

In traditional encryption schemes, security is based on an idealized assumption that the adversary cannot get any information about the private keys and internal state. However, the practice shows this assumption is quite invalid. Many cryptographic schemes are vulnerable to side-channel attacks, where an adversary can learn meaningful information about a system by using some of the physical information that the algorithm outputs, such as running time, power consumption, and fault detection etc.. In order to characterize the leaked information that the adversary knows in the system better, various leakage models are presented. Some of them are motivated by practical issues, while others are for theoretical needs.

- The only computation leaks information model was proposed by Micali [14]. It requires that the leak only occurs in the memory part of the system executing the calculation, and the memory part that does not participate in the calculation is not leaked. The reason given in [14] is as "data can be placed in some form of storage where, when not being accessed and computed upon, it is totally secure."

- The relative leakage model (also named memory-attacks model) was proposed by Alwen *et al.* [1] to deal with cold boot attacks where the part not involved in the operation also leaked information. In this model, the leakage amount is bounded by a predetermined value.

- The bounded retrieval model is a model that stronger than the relative leakage model [2, 21, 25, 27, 29]. In this model, the leakage parameter l is an arbitrary and independent parameter of the system, and secret keys can be increased to allow l bits of leakage without affecting the size of public keys.

- The continuous leakage model [30] was put forward to solve the situation that the leakage bits exceed the predetermined number in which the leakage is unbounded in the lifecycle of the system, but it is bounded between consecutive updates.

- The auxiliary input model was presented by Dodis [8] which required that polynomial time adversary cannot recover sk from $f(sk)$ with negligible probability. Meanwhile, Yuen *et al.* [22] proposed a model which combined the concepts of the auxiliary inputs and continual memory leakage. The scheme [28] also comes from this model.

While the ABE schemes constructed in the above leakage model cannot achieve anonymity except [25, 27]. Additionally, the number of leakage bits in [25] is bounded

and the performance of [27] is inefficient because its decryption time depends not only on the leakage parameter but also on the number of attributes. Hence, it is natural to ask whether there is an efficient anonymous leakage-resilient ciphertext-policy attribute-based encryption scheme resilient to the continual leakage.

1.1 Related Work

ABE [18] was regarded as a highly promising public key primitive for realizing scalable and fine-grained access control systems. Goyal *et al.* [9] formulated the idea of ABE and presented key-policy ABE (KP-ABE). Then Bethencourt *et al.* [4] put forward to the ciphertext-policy ABE (CP-ABE). In KP-ABE, the private keys are associated with access policies and ciphertexts are labeled with sets of attributes. While in CP-ABE, ciphertexts are associated with access policies and private keys are labeled with sets of attributes. ABE has become a research hotspot because it implements one-to-many encryption. Following this trend, many schemes have been proposed [5, 7, 13, 26].

Schemes [4,9] adopted the access tree which is a monotone access structure. A KP-ABE scheme can handle non-monotone access structures over attributes where the access structures can be a boolean formula involving AND, OR, NOT, and threshold operations was proposed by R. Ostrovsky *et al.* [16]. Lewko *et al.* [12] first put forward a CP-ABE scheme and a KP-ABE scheme where the access structures were monotone span program (MSP). Both schemes are proven to be fully secure under Decisional Subgroup assumptions in the standard model over composite order bilinear group. Subsequently, Okamoto and Takashima [15] brought forward a KP-ABE and a CP-ABE for non-monotone access structures which were shown to be fully secure under a standard assumption, the Decisional Linear (DLIN) assumption, in the standard model over prime order bilinear group. Then Waters [19] proposed three efficient selectively secure CP-ABE constructions by employing MSP to express access structure in the standard model under Decisional Parallel Bilinear Diffie-Hellman Exponent (PBDHE) assumption. Lately, Attrapadung *et al.* [3] introduced a KP-ABE scheme for non-monotone access structures with constant size ciphertexts, which was selectively secure under Decisional q-parallel Bilinear Diffie-Hellman Exponent (q-DBDHE) assumption in the standard model over prime order bilinear group. This scheme also adopted the method of Ostrovsky *et al.* [16] to convert the non-monotone access structures to monotone access structures with negative attributes. Based on the fact that there are some monotone access structures for which the size of MSP is at least the number of minimal sets, while the number of minimal sets is constant. Pandit and Barua [17] put forward an ABE which used minimal sets to describe general access structures. They also constructed a corresponding hierarchical (H)KP-ABE scheme. All of the schemes achieve full security in the standard model over composite order bilinear group.

Though ABE can be directly applied to design secure access control, there is an increasing need to protect user's privacy in access control systems. In order to address the problem, the concept of anonymous ABE was introduced in schemes [10,11]. More related works are referred to [23, 24, 27]. In the anonymous CP-ABE, ciphertexts can not reveal the information of corresponding attributes in the access policies. A user obtains his/her secret keys if the corresponding attribute sets satisfy the access structure embedded in the ciphertexts. The user cannot decrypt and guess what access policy was specified by the data owner. As we know, there is no efficient anonymous CP-ABE can achieve constant size ciphertexts and adaptive leakage-resilient security in the standard model.

1.2 Our Contributions

Based on the works of [30] and [6], we put forward efficient anonymous leakage-resilient CP-ABE schemes for general access structures, which has better leakage rate and adaptive security under the four static assumptions in the standard model over composite order bilinear group. In addition, the proposed schemes were built on the relative leakage model, and implicitly used an update algorithm to tolerate continual leakage on the private keys. In the security proof, we use the dual system encryption [20], where we extend the semi-functional keys into two types: truly semi-functional and nominally semi-functional. Normal keys and nominally semi-functional keys can decrypt normal ciphertexts and semi-functional ciphertexts, but truly semi-functional keys cannot decrypt the challenge semi-functional ciphertexts. In addition, the method to prove the indistinguishability of nominally semi-functional and truly semi-functional is similar to [30], so we omitted it in the paper. The access structure used in the first construction is constructed by minimal sets with multi-valued attributes which provides the ability to fast decryption.

1.3 Organizations

The rest of paper is organized as follows. In Section 2, some preliminaries are given. Section 3 gives the definition of leakage resilience of ABE. The security definition is presented in Section 4. The construction of scheme and anonymity, performance, and efficiency analysis are given in Section 5. And security proof is introduced in Section 6. In Section 7, we give an anonymous leakage-resilient CP-ABE scheme for non-monotone access structures. Finally, we conclude this paper in Section 8.

2 Preliminaries

2.1 Notations

1) Angle brackets $\langle \cdot, \cdot \rangle$ denotes two vectors inner product, and parentheses $(\cdot, \cdot, \cdot)$ denotes vectors. The dot

product of vectors is denoted by '$\cdot$' and component-wise multiplication is denoted by '$*$'.

2) The fact that χ is picked uniformly at random from a finite set Ω is denoted by $\chi \xleftarrow{\$} \Omega$, and that all ψ, ω, ζ are picked independently and uniformly at random from Ω is denoted by $\psi, \omega, \zeta \xleftarrow{\$} \Omega$.

3) Let $\vec{\rho} = (\rho_1, \rho_2, ..., \rho_n), \vec{\sigma} = (\sigma_1, \sigma_2, ..., \sigma_n)$, $g^{\vec{\rho}}$ denote the vector of group element $g^{\vec{\rho}} = (g^{\rho_1}, g^{\rho_2}, ..., g^{\rho_n})$, the inner product vectors $\vec{\rho}$ and vector $\vec{\sigma}$ is denoted by $\langle \vec{\rho}, \vec{\sigma} \rangle$ and the bilinear group inner product is denoted by $\hat{e}_n(g^{\vec{\rho}}, g^{\vec{\sigma}})$. *i.e.*, $\langle \vec{\rho}, \vec{\sigma} \rangle = \sum_{i \in [n]} \rho_i \sigma_i$, and $\hat{e}_n(g^{\vec{\rho}}, g^{\vec{\sigma}}) = \prod_{i \in [n]} \hat{e}(g^{\rho_i}, g^{\sigma_i}) = \hat{e}(g, g)^{\langle \vec{\rho}, \vec{\sigma} \rangle}$.

4) A negligible function of λ is denoted by $negl(\lambda)$.

5) $\{1, 2, ..., n\}$ is denoted by $[n]$.

2.2 Minimal Set and Its Critical Set

Definition 1. *Let Γ be a monotonic access structure over the set of attributes $V = \{a_1, a_2, ..., a_n\}$. If $\forall A \in \Gamma \backslash \{B\}$ where $B \in \Gamma$, we have $A \subset B$ invalid, then B is called a minimal authorized set. The collection of all minimal sets in Γ is called the basis of Γ.*

Definition 2. ***(Dual of access structure)*** *If $V \backslash A = A^c \notin \Gamma$ where $A \subset V$, then the collection of sets A is composed of the dual of access structure $\Gamma^{\perp}$ of an access structure Γ over V.*

Definition 3. ***(Critical set of minimal sets)*** *If every $Y_i \in \mathcal{H}$ contains a set $B_i \subset Y_i$, there is $|B_i| \geq 2$:*

The set B_i uniquely determines Y_i in the set $\mathcal{H}$. i.e., *no other set in $\mathcal{H}$ contains B_i.*

$\forall Z \subset B_i$, set $S_Z = \bigcup_{Y_j \in \mathcal{H}, Y_j \bigcap Z \neq \emptyset} (Y_j \backslash Z)$ does not contain any element of $\mathcal{B}$.

If (I) and (II) hold, where $\mathcal{B} = \{Y_1, Y_2, ..., Y_r\}$ is the set of minimal set of an access structure Γ, and $\mathcal{H} \in \mathcal{B}$ be a subset of minimal sets, then $\mathcal{H}$ is called a critical set of minimal sets for $\mathcal{B}$.

2.3 Complexity Assumptions

Assumption 1. Pick $g_1 \xleftarrow{\$} \mathbb{G}_{p_1}, g_3 \xleftarrow{\$} \mathbb{G}_{p_3}, g_4 \xleftarrow{\$} \mathbb{G}_{p_4}$, $T_1 \xleftarrow{\$} \mathbb{G}_{p_1 p_4}, T_2 \xleftarrow{\$} \mathbb{G}_{p_1 p_2 p_4}$ and set $E = (\mathbb{G}, g_1, g_3, g_4)$. Define the advantage of an algorithm $\mathcal{A}$ in breaking Assumption 1 to be

$$Adv^1_{\mathcal{A}}(\lambda) = |Pr[\mathcal{A}(E, T_1) = 1] - Pr[\mathcal{A}(E, T_2) = 1]| \quad (1)$$

We say that Assumption 1 holds if for all PPT algorithm $\mathcal{A}$, $Adv^1_{\mathcal{A}}(\lambda) \leq negl(\lambda)$ holds for the security λ.

Assumption 2. Pick $g_1, U_1 \xleftarrow{\$} \mathbb{G}_{p_1}, U_2, W_2 \xleftarrow{\$} \mathbb{G}_{p_2}$, $g_3, W_3 \xleftarrow{\$} \mathbb{G}_{p_3}, g_4 \xleftarrow{\$} \mathbb{G}_{p_4}, T_1 \xleftarrow{\$} \mathbb{G}_{p_1 p_2 p_3}, T_2 \xleftarrow{\$} \mathbb{G}_{p_1 p_3}$ and set $E = (\mathbb{G}, g_1, g_3, g_4, U_1 U_2, W_2 W_3)$. Define the advantage of an algorithm $\mathcal{A}$ in breaking Assumption 2 to be

$$Adv^2_{\mathcal{A}}(\lambda) = |Pr[\mathcal{A}(E, T_1) = 1] - Pr[\mathcal{A}(E, T_2) = 1]| \quad (2)$$

We say that Assumption 2 holds if for all PPT algorithm $\mathcal{A}$, $Adv^2_{\mathcal{A}}(\lambda) \leq negl(\lambda)$ holds for the security λ.

Assumption 3. Pick $\alpha, s, r \xleftarrow{\$} \mathbb{Z}_N, g_1 \xleftarrow{\$} \mathbb{G}_{p_1}, g_4 \xleftarrow{\$} \mathbb{G}_{p_4}$, $U_2, W_2, g_2 \xleftarrow{\$} \mathbb{G}_{p_2}, g_3 \xleftarrow{\$} \mathbb{G}_{p_3}, T_1 = \hat{e}(g_1, g_1)^{\alpha s}, T_2 \xleftarrow{\$} \mathbb{G}_T$, and set $E = (\mathbb{G}, g_1, g_2, g_3, g_4, g_2^r, U_2^r, g_1^{\alpha} U_2, g_1^s W_2)$. Define the advantage of an algorithm $\mathcal{A}$ in breaking Assumption 3 to be

$$Adv^3_{\mathcal{A}}(\lambda) = |Pr[\mathcal{A}(E, T_1) = 1] - Pr[\mathcal{A}(E, T_2) = 1]| \quad (3)$$

We say that Assumption 3 holds if for all PPT algorithm $\mathcal{A}$, $Adv^3_{\mathcal{A}}(\lambda) \leq negl(\lambda)$ holds for the security λ.

Assumption 4. Pick $s, \hat{r}, \hat{s} \xleftarrow{\$} \mathbb{Z}_N$, $g_1, U_1 \xleftarrow{\$} \mathbb{G}_{p_1}$, $g_4, U_4 \xleftarrow{\$} \mathbb{G}_{p_4}, U_2, W_2, g_2 \xleftarrow{\$} \mathbb{G}_{p_2}, g_3 \xleftarrow{\$} \mathbb{G}_{p_3}, W_{24}, D_{24} \xleftarrow{\$} \mathbb{G}_{p_2 p_4}, T_1 \xleftarrow{\$} U_1^s D_{24}, T_2 \xleftarrow{\$} \mathbb{G}_{p_1 p_2 p_4}$, and set $E = (\mathbb{G}, g_1, g_2, g_3, g_4, U_1 U_4, U_1^{\hat{r}} U_2, g_1^{\hat{r}} W_2, g_1^s W_{24}, U_1 g_3^{\hat{s}})$. Define the advantage of an algorithm $\mathcal{A}$ in breaking Assumption 4 to be

$$Adv^4_{\mathcal{A}}(\lambda) = |Pr[\mathcal{A}(E, T_1) = 1] - Pr[\mathcal{A}(E, T_2) = 1]| \quad (4)$$

We say that Assumption 4 holds if for all PPT algorithm $\mathcal{A}$, $Adv^4_{\mathcal{A}}(\lambda) \leq negl(\lambda)$ holds for the security λ.

3 Leakage Resilience of CP-ABE

A CP-ABE scheme with continual leakage model is composed of the following five algorithms:

Setup$((\lambda, V, l) \rightarrow (PK, MSK))$**:** The setup algorithm takes a security parameter λ, a description of attribute universe set V and a leakage bound l as input. It outputs system public keys PK and master secret keys MSK.

KeyGen$((PK, MSK, S) \rightarrow SK_S)$**:** The key generation algorithm inputs the public keys PK, master secret keys MSK and an attribute set S, returns secret keys SK_S.

UpdateUSK$((PK, S, SK_S) \rightarrow SK'_S)$**:** On input the public keys PK, a set of attributes S and the secret keys SK_S, it outputs re-randomized secret keys SK'_S.

Encrypt$((PK, M, \Gamma) \rightarrow CT_\Gamma)$**:** The encryption algorithm takes the public keys PK, a message M, and an access structure Γ over the universe of attributes as input, and outputs ciphertexts CT_Γ such that only users whose attribute sets satisfy the access structure Γ should be able to extract M.

Decrypt$((PK, CT_\Gamma, SK_S) \rightarrow M)$**:** The algorithm takes the public keys PK, ciphertexts CT_Γ and secret keys SK_S as input, outputs the message M if and only if the attribute set S of key SK_S satisfies the access structure Γ.

4 Security Definition

For key leakage attacks, we provide a game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ to achieve an anonymous leakage-resilient ciphertext-policy attribute-based encryption scheme. The security parameter and the upper bound of leakage are denoted by λ and l respectively.

Setup: The challenger $\mathcal{C}$ runs the setup algorithm to generate the public keys and master keys (PK, MSK), and sends the public keys to the adversary $\mathcal{A}$ while keeps the master secret keys. At the same time, $\mathcal{C}$ creates two initial empty lists: $\mathcal{Q} = (hd, S, SK_S, L_{SK})$, $\mathcal{R} = (hd, S)$ to store records, where all records are associated with a handle hd and L_{SK} means total leakage bits.

Phase 1: In this stage, $\mathcal{A}$ can adaptively perform the following queries:

- *Key Generation queries*: $\mathcal{A}$ submits the attribute set S to $\mathcal{C}$, and $\mathcal{C}$ runs the key generation algorithm to generate the private keys SK_S. Challenger sets $hd = hd + 1$, then adds $(hd, S, SK_S, 0)$ to the set $\mathcal{Q}$. In this query, $\mathcal{C}$ only gives $\mathcal{A}$ hd of the generated keys rather than the concrete keys itself.
- *Leakage queries*: $\mathcal{A}$ gives a polynomial-time computable arbitrary function $f : \{0,1\}^\star \rightarrow \{0,1\}^\star$ with a queried handle hd of the keys to $\mathcal{C}$. $\mathcal{C}$ finds the tuple (hd, S, SK_S, L_{SK}) and checks if $L_{SK} + |f(SK)| \leq l$ established. If this is true, it returns $f(SK)$ to $\mathcal{A}$ and updates L_{SK} with $L_{SK} + |f(SK)|$. If the check fails, it returns $\perp$ to the $\mathcal{A}$.
- *Reveal queries*: $\mathcal{A}$ gives the handle hd for a specified key SK_S to $\mathcal{C}$. The $\mathcal{C}$ scans $\mathcal{Q}$ to find the requested entry and returns the private keys SK_S to $\mathcal{A}$. Then $\mathcal{C}$ removes the item from the set $\mathcal{Q}$ and adds it to the set $\mathcal{R}$.
- *Update queries*: $\mathcal{A}$ issues a key update query for SK_S. $\mathcal{C}$ searches the record in $\mathcal{Q}$. If it is not found, $\mathcal{C}$ returns the keys with key generation algorithm and sets $L_{SK} = 0$. Otherwise, $\mathcal{C}$ returns with UpdateUSK algorithm and updates the corresponding $L_{SK} = 0$.

Challenge: $\mathcal{A}$ outputs two pairs of message and access structure (M_0, Γ_0), (M_1, Γ_1) to $\mathcal{C}$, where for every $S \in \mathcal{R}$, neither satisfies Γ_0 nor satisfies Γ_1. With the restriction that the length of the message M_0 equals to the length of the message M_1, $\mathcal{C}$ selects $b \in \{0,1\}$ randomly and encrypts the message M_b under the access structure Γ_b, and sends the resulting ciphertexts to $\mathcal{A}$.

Phase 2: This phase is the same as Phase 1 with the additional restrictions that reveal queries and update queries can be performed, and cannot execute leakage queries. The attributes of the query do not satisfy the challenge access structure.

Guess: $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$ and wins the game if $b' = b$.

We say that an anonymous attribute-based encryption scheme is l leakage-resilient and adaptively secure against chosen plaintext attacks (ANON-IND-CPA) if for all polynomial time adaptive adversaries $\mathcal{A}$, the advantage of $\mathcal{A}$ in the above mentioned game is negligible, where the advantage of $\mathcal{A}$ is defined as $Adv_{\mathcal{A}}^{ANON-IND-CPA}(\lambda, l) = |Pr[b' = b] - \frac{1}{2}|$.

5 Anonymous Leakage-Resilient CP-ABE for MAS

5.1 Concrete Construction

Our scheme relies on a composite order bilinear group where its order is $N = p_1p_2p_3p_4$, and p_1, p_2, p_3, p_4 are distinct primes. The main system is built in $\mathbb{G}_{p_1}$ subgroup, while the subgroup $\mathbb{G}_{p_2}$ acts as the semi-functional space. The subgroup $\mathbb{G}_{p_3}$ provides the additional randomness on keys to isolate keys in our hybrid games. $\mathbb{G}_{p_4}$ will make the scheme achieve anonymity. Then we would extend the composite order group to multiple dimensional to tolerate the possible leakage.

Let $V = \{attr_1, attr_2, ..., attr_n\}$ be a set of attributes. Each attribute contains n_i possible values and $v_{i,j}$ represents the jth value of $attri$, and $I \subset \{1, 2, ..., n\}$ is the attribute name index.

Setup$((\lambda, V, l) \rightarrow (PK, MSK))$**:** The setup algorithm takes as input a security parameter λ, the attribute universe description V and a leakage upper bound l. Then the algorithm generates the public keys and master secret keys as follows. Run the bilinear group generator to produce $\Phi = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e})$. Define $negl = p_2^{-\tau}$ as the allowable maximum probability in succeeding in leakage guess and compute $\omega = \lceil 1 + 2\tau + \frac{l}{\log p_2} \rceil$, where τ is a positive constant. In practice, $\omega \approx \lceil 1 + \frac{l}{\log p_2} \rceil$. Select $g_1, X_1 \in \mathbb{G}_{p_1}$, $g_3 \in \mathbb{G}_{p_3}$, $g_4, X_4 \in \mathbb{G}_{p_4}$, $\alpha \in \mathbb{Z}_N$, $\vec{\rho} \in \mathbb{Z}_N^\omega$ randomly. For each $i \in [n], j \in [n_i]$, choose random values $t_{i,j} \in \mathbb{Z}_N$ and set the public keys as follows.

$$PK = (N, g_1, g_4, g_1^{\vec{\rho}}, y, Y, T_{i,j}; \forall i \in [n], j \in [n_i]),$$

where

$$y = \hat{e}(g_1, g_1)^{\alpha}, Y = X_1 X_4, T_{i,j} = g^{t_{i,j}}.$$

The master secret keys are

$$MSK = (X_1, g_3, \alpha).$$

KeyGen$((PK, MSK, S) \rightarrow SK_S)$**:** This algorithm takes PK, MSK and an attribute set $S = \{v_{1,x_1}, v_{2,x_2}, ..., v_{n',x'_n}\}$ as input, where $n' \leq n, 1 \leq x_i \leq n_i$ for each $1 \leq i \leq n'$. Then the algorithm chooses random values $t, y_2, y_3 \in \mathbb{Z}_N, \vec{y}_1, \vec{\sigma} \in \mathbb{Z}_N^{\omega}$, and $y_{i,j} \in \mathbb{Z}_N$ for $v_{i,j} \in S$, and outputs the secret keys as follows.

$$SK_S = (S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S),$$

in which

$$\vec{K}_1 = g_1^{\vec{\sigma}} * g_3^{\vec{y}_1}, \qquad K_3 = g_1^t g_3^{y_3},$$
$$K_2 = g_1^{\alpha + \langle \vec{\rho}, \vec{\sigma} \rangle} X_1^t g_3^{y_2}, \qquad K_{i,j} = T_{i,j}^t g_3^{y_{i,j}}.$$

UpdateUSK$((PK, S, SK_S) \rightarrow SK'_S)$**:** The key update algorithm selects $\Delta t, \Delta y_2, \Delta y_3 \in \mathbb{Z}_N, \Delta \vec{y}_1, \Delta \vec{\sigma} \in \mathbb{Z}_N^{\omega}$ randomly, picks $\Delta y_{i,j} \in \mathbb{Z}_N$ for $v_{i,j} \in S$ at random, and outputs a new key SK'_S:

$$SK'_S = (S, \vec{K}'_1, K'_2, K'_3, K'_{i,j}; \forall v_{i,j} \in S),$$

where

$$\vec{K}'_1 = \vec{K}_1 * g_1^{\Delta \vec{\sigma}} * g_3^{\Delta \vec{y}_1}, \qquad K'_3 = K_3 g_1^{\Delta t} g_3^{\Delta y_3},$$
$$K'_2 = K_2 g_1^{\langle \vec{\rho}, \Delta \vec{\sigma} \rangle} X_1^{\Delta t} g_3^{\Delta y_2}, \qquad K'_{i,j} = K_{i,j} T_{i,j}^{\Delta t} g_3^{\Delta y_{i,j}}.$$

Encrypt$((PK, M, \Gamma) \rightarrow CT_{\Gamma})$**:** At first, this algorithm converts the monotonic access structure Γ to the set of minimal sets $\mathcal{B} = \{B_1, B_2, ..., B_{\tilde{m}}\}$, where $B_k (k \in [\tilde{m}])$ is a set of attribute values. It selects $s, s_1, s_2, ..., s_{\tilde{m}} \in \mathbb{Z}_N, \vec{d} \in \mathbb{Z}_N^{\omega}, W_k, V_k \in \mathbb{G}_{p_4} (k \in [\tilde{m}])$ randomly, then outputs the resulting ciphertexts CT_{Γ} and the index $I_{B_k} \subset \{1, 2, ..., n\} (k \in [\tilde{m}])$ corresponding to attribute $B_k (k \in [\tilde{m}])$.

$$CT_{\Gamma} = (\{I_{B_k}\}_{k \in [\tilde{m}]}, C_0, \vec{C}_1, C_2, \vec{C}_3, \vec{C}_4),$$

where

$$C_0 = M y^s, \qquad \vec{C}_1 = g_1^{s\vec{\rho}} * g_4^{\vec{d}},$$
$$C_2 = g_1^s g_4, \qquad \vec{C}_4 = (C_{4,k})_{k \in [\tilde{m}]} = (g_1^{s_k} V_k)_{k \in [\tilde{m}]},$$
$$\vec{C}_3 = \{C_{3,k}\}_{k \in [\tilde{m}]} = \{Y^s (\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k} W_k\}_{k \in [\tilde{m}]}.$$

Decrypt$((PK, CT_{\Gamma}, SK_S) \rightarrow M)$**:** If the attributes set S satisfies the access structure specified by $\mathcal{B}$, then S must be a superset of a minimal set in $\mathcal{B}$. Let $B_k \subset S$ for some $k \in [\tilde{m}]$, this algorithm calculates

$$M = \frac{C_0 \cdot \hat{e}_{\omega}(\vec{C}_1, \vec{K}_1) \hat{e}(C_{3,k}, K_3)}{\hat{e}(C_2, K_2) \hat{e}(C_{4,k}, \prod_{v_{i,j} \in B_k} K_{i,j})}$$

5.2 Correctness

The correctness can be checked by applying the orthogonality of $\mathbb{G}_{p_i}$, where $i = 1, 2, 3, 4$. If the attributes set S satisfies the access structure specified by $\mathcal{B}$, then one can obtain the below equations hold.

$$\begin{aligned}
\hat{e}_{\omega}(\vec{C}_1, \vec{K}_1) &= \hat{e}_{\omega}(g_1^{s\vec{\rho}} * g_4^{\vec{d}}, g_1^{\vec{\sigma}} * g_3^{\vec{y}_1}) \\
&= \hat{e}_{\omega}(g_1^{s\vec{\rho}}, g_1^{\vec{\sigma}}) \\
&= \hat{e}(g_1, g_1)^{s\langle \vec{\rho}, \vec{\sigma} \rangle} \\
\hat{e}(C_2, K_2) &= \hat{e}(g_1^s g_4, g_1^{\alpha + \langle \vec{\rho}, \vec{\sigma} \rangle} X_1^t g_3^{y_2}) \\
&= \hat{e}(g_1, g_1)^{\alpha s + s\langle \vec{\rho}, \vec{\sigma} \rangle} \hat{e}(g_1, X_1)^{st} \\
\hat{e}(C_{3,k}, K_3) &= \hat{e}(Y^s (\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k} W_k, g_1^t g_3^{y_3}) \\
&= \hat{e}(g_1, Y)^{st} \hat{e}(\prod_{v_{i,j} \in B_k} T_{i,j}, g_1)^{s_k t} \\
\hat{e}(C_{4,k}, \prod_{v_{i,j} \in B_k} K_{i,j}) &= \hat{e}(g_1^{s_k} V_k, \prod_{v_{i,j} \in B_k} T_{i,j}^t g_3^{y_{i,j}}) \\
&= \hat{e}(g_1, \prod_{v_{i,j} \in B_k} T_{i,j})^{s_k t}
\end{aligned}$$

5.3 Anonymity Analysis

This section will show that the proposed scheme achieves the anonymity over the composite order bilinear group. Compared with scheme [30], our scheme adds some random elements in $\mathbb{G}_{p_4}$ to each part of the ciphertexts. These random elements will not make an effect on the decryption process. However, they are necessary for anonymity of the scheme. Because if there is no such elements, for some minimal sets B_k^*, the adversary may determine whether the ciphertext component $C_{3,k}$ of the ciphertext $\vec{C}_3$ is encrypted under B_k^* or not. In our scheme, by utilizing the DDH-test $\hat{e}(C_{3,k}, g_1) \stackrel{?}{=} \hat{e}(Y, C_2) \hat{e}(\prod_{v_{i,j^*} \in B_k^*} T_{i,j^*}, C_{4,k})$ to determine whether the ciphertext component $C_{3,k}$ is encrypted under the B_k^* or not. The DDH-test $\hat{e}(C_{3,k}, g_1) \stackrel{?}{=} \hat{e}(Y, C_2) \hat{e}(\prod_{v_{i,j^*} \in B_k^*} T_{i,j^*}, C_{4,k})$ is the same as $\frac{\hat{e}(C_{3,k}, g_1)}{\hat{e}(Y, C_2) \hat{e}(\prod_{v_{i,j^*} \in B_k^*} T_{i,j^*}, C_{4,k})} \stackrel{?}{=} 1$. The followings are the detailed analyses.

$$\begin{aligned}
\hat{e}(C_{3,k}, g_1) &= \hat{e}(Y^s (\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k} W_k, g_1) \\
&= \hat{e}(Y^s, g_1) \hat{e}((\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k}, g_1) \\
&= \hat{e}(Y, g_1)^s \hat{e}(\prod_{v_{i,j} \in B_k} T_{i,j}, g_1)^{s_k} \\
\hat{e}(Y, C_2) &= \hat{e}(Y, g_1)^s \hat{e}(Y, g_4)
\end{aligned}$$

$$\begin{aligned}
\hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, C_{4,k}) &= \hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, g_1^{s_k} V_k) \\
&= \hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, g_1^{s_k}) \\
&= \hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, g_1)^{s_k} \\
\frac{\hat{e}(C_{3,k}, g_1)}{\hat{e}(Y, C_2)\hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, C_{4,k})} &= \frac{\hat{e}(\prod_{v_{i,j}\in B_k} T_{i,j}, g_1)^{s_k}}{\hat{e}(Y, g_4)\hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, g_1)^{s_k}}
\end{aligned}$$

If $B_k = B_k^*$, then $v_{i,j} = v_{i,j^*}$ for all i, $1 \le i \le n' \le n$. Therefore $\frac{\hat{e}(C_{3,k}, g_1)}{\hat{e}(Y, C_2)\hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, C_{4,k})} = \frac{1}{\hat{e}(Y, g_4)}$.

If $B_k \neq B_k^*$, then there exists at last one k', where $1 \le k' \le n' \le n$ such that $v_{k',j} \neq v_{k',j^*}$. Without loss of generality, let $v_{k',j} = v_{k',j^*}$, for all $1 \le i \le n' \le n$ except $i = k'$. Then $t_{i,j} = t_{i,j^*}$. Therefore $\frac{\hat{e}(C_{3,k}, g_1)}{\hat{e}(Y, C_2)\hat{e}(\prod_{v_{i,j^*}\in B_k^*} T_{i,j^*}, C_{4,k})} = \frac{\hat{e}(T_{k',j}, g_1)^{s_k}}{\hat{e}(Y, g_4)\hat{e}(T_{k',j^*}, g_1)^{s_k}}$.

In both cases, $B_k = B_k^*$ and $B_k \neq B_k^*$, the DDH-test gives a random element of $\mathbb{G}_T$ so that the adversary will be not able to determine whether the component $C_{3,k}$ of the ciphertext $\vec{C_3}$ is encrypted under the B_k^* or not. So the access structure is hidden, and our scheme is anonymous.

5.4 Performance Analysis

As shown in Table 1, we give the performance comparisons of schemes [21,27,30] and the proposed scheme in the access policy, leakage model, support multi-functionality and anonymity. All these schemes are constructed under the key leakage model, in which the access structure of [21,27] is denoted by the linear secret sharing (LSSS), while that of [30] and the proposed scheme are represented by the minimal sets. In addition, it can be found that [21,30] do not support anonymity and scheme [21,27] do not support multi-show functionality. However, our scheme achieves the anonymity and attribute multi-show ability simultaneously.

5.5 Efficiency Analysis

We present the performance evaluation based on our DMA implementation prototype. Our experiment is implemented on Pairing-Based Cryptography (PBC) library to implement the scheme. We will compare the computational efficiency of our scheme with scheme [21,27,30]. In the Figures 1, 2 and 3, the leakage parameter is set to be $\omega = 5$.

Figure 1 shows the comparison of key generation time with different number of attributes, where the number of attribute changes from 10 to 50.

Figure 2 presents the comparison of update time with different number of attributes.

Figure 3 provides the comparison of encryption time with different number of minimal sets, where the number of minimal sets varies from 5 to 25.

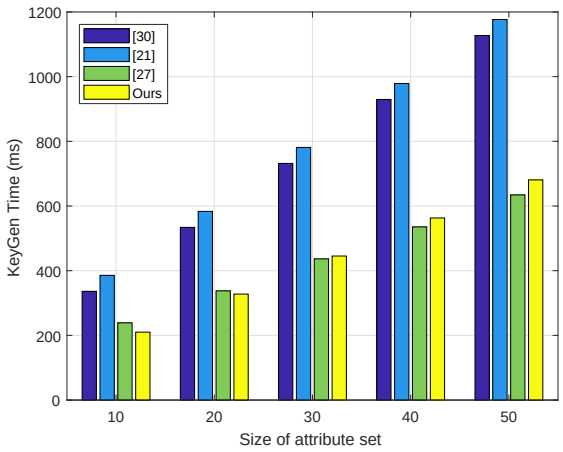

Figure 1: KeyGen time with different number of attributes

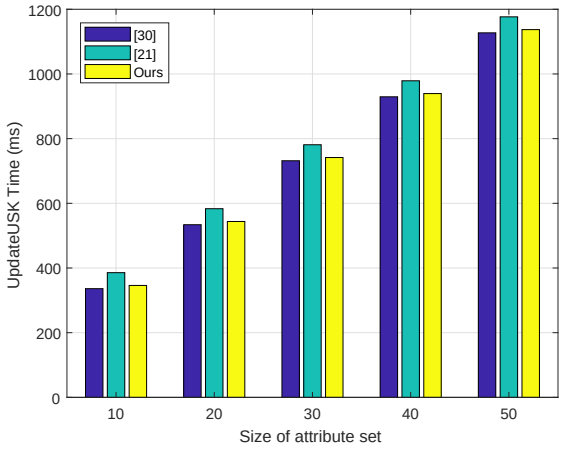

Figure 2: UpdateUSK time with different number of attributes

Figure 4 gives the comparison of decryption time with different leakage parameters, where the leakage parameter changes from 5 to 25.

From the analysis of the experimental results, we can see that our scheme has advantages over [27] when the number of attributes is between 10 and 20. While, compared with [21,30], the proposed scheme spends less time. Moreover, the proposed scheme has obvious advantage over [27] in decryption. In summary, our scheme is quite practical and efficient.

6 Security Proof

In the proof, we generate normal private keys and ciphertexts which are used in the real scheme. Then we generate semi-functional keys and ciphertexts which are used in the proofs. They are shown as follows.

- **KeyGenSF**. $SK_S = (S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S)$ be the normal keys. The semi-functional keys are as follows.

Table 1: Performance analysis

Scheme	Access policy	Leakage Model	Multi-show attr	Anonymity
[21]	LSSS	Continual leakage	No	No
[30]	Minimal Sets	Continual leakage	Yes	No
[27]	LSSS	Bounded leakage	Not	Yes
Ours	Minimal Sets	Continual leakage	Yes	Yes

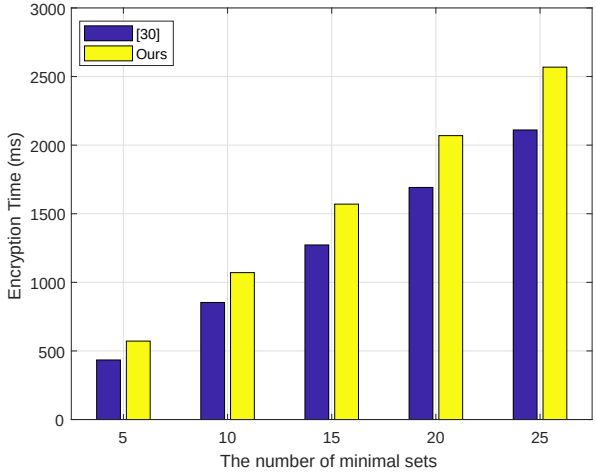

Figure 3: Encryption time with different number of minimal sets

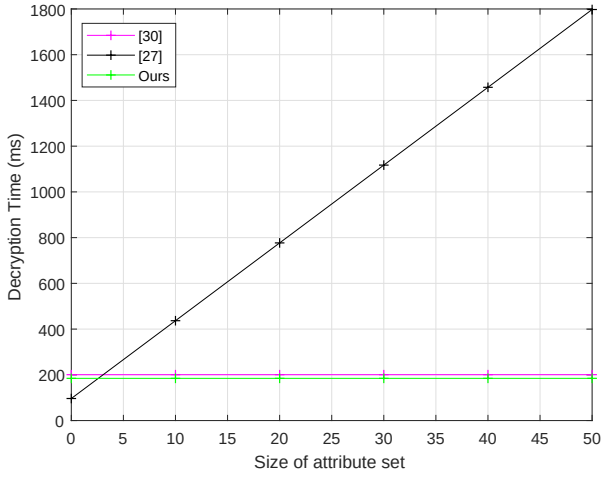

Figure 4: Decryption time with different number of attributes

  - Type 1: $S\bar{K}_S = (S, \vec{K}_1 * g_2^{\vec{d}_1}, K_2 g_2^{d_2}, K_3 g_2^{d_3}, K_{i,j} g_2^{d_{i,j}}; \forall v_{i,j} \in S)$, where g_2 is a generator of group $\mathbb{G}_{p_2}$ and $\vec{d}_1, d_2, d_3, d_{i,j}$ are random elements in $\mathbb{Z}_N$.
  - Type 2: $S\bar{K}_S = (S, \vec{K}_1, K_2 g_2^{d_2}, K_3, K_{i,j}; \forall v_{i,j} \in S)$

- **EncSF**. Let $CT_\Gamma = (\{I_{B_k}\}_{k\in[\tilde{m}]}, C_0, \vec{C}_1, C_2, \vec{C}_3, \vec{C}_4)$ be a normal ciphertext. The semi-functional ciphertexts are converted as: $C\bar{T}_\Gamma = (\{I_{B_k}\}_{k\in[\tilde{m}]}, C_0, \vec{C}_1 * g_2^{\vec{e}_1}, C_2 g_2^{e_2}, \vec{C}_3 * g_2^{\vec{e}_3}, \vec{C}_4)$, where $\vec{e}_1, e_2, \vec{e}_3$ are random elements in $\mathbb{Z}_N$.

If we use the Type 1 of semi-functional keys to decrypt a semi-functional ciphertext, we will obtain extra term $\hat{e}(g_2, g_2)^{\langle \vec{d}_1, \vec{e}_1\rangle - d_2 e_2 + d_3 e_{3,k}}$. If $\langle \vec{d}_1, \vec{e}_1\rangle - d_2 e_2 + d_3 e_{3,k} = 0$, we can call it a nominally semi-functional key, otherwise it is truly semi-functional.

The proof uses a series of indistinguishable games to prove the indistinguishability between in the $Game_{real}$ and $Game_{final1}$. There will be $2Q+4$ games between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, where Q is the number of key queries times and k' is from 0 to Q. The concrete definition of games is as follows.

$Game_{real}$: This is the real anonymous ABE security game, where all private keys and the ciphertexts are in normal form.

$Game_0$: All private keys are in normal form, and the challenge ciphertexts are in semi-functional form.

$Game_{k',1}$: The challenge ciphertexts are semi-functional. The first $k'-1$ keys are semi-functional of Type 2, and the k'^{th} key is the semi-functional form of Type 1. The remaining keys are normal.

$Game_{k',2}$: The challenge ciphertexts are semi-functional. The first k' keys are semi-functional of Type 2, and the remaining keys are normal.

$Game_{final0}$: All private keys are the Type 2 of semi-functional keys. And the challenge ciphertexts are semi-functional where C_0 is random in group $\mathbb{G}_T$.

$Game_{final1}$: It is same as $Game_{final0}$ except that the $\vec{C}_3$ is random in group $\mathbb{G}_{p_1 p_2 p_4}$.

We can see that in $Game_{Q,2}$, all of the keys are semi-functional. And in the last game, the adversary has no advantage.

Lemma 1. *Suppose there exists a PPT adversary $\mathcal{A}$ who can distinguish $Game_{real}$ and $Game_0$ with the non-negligible advantage ϵ, then there is a PPT algorithm $\mathcal{B}$ with the advantage ϵ in breaking Assumption 1.*

Proof. $\mathcal{B}$ receives $E = (\mathbb{G}, g_1, g_3, g_4)$ from the challenger $\mathcal{C}$ and simulates $Game_{real}$ or $Game_0$ with $\mathcal{A}$ depending on whether $T \xleftarrow{\$} \mathbb{G}_{p_1 p_4}$ or $T \xleftarrow{\$} \mathbb{G}_{p_1 p_2 p_4}$. □

Setup: $\mathcal{B}$ takes the security parameter λ and leakage upper bound l as input and outputs the description of group: $\Phi = (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e})$. Then $\mathcal{B}$ generates the public keys as follows. Set $\omega = \lceil 1 + \frac{l}{logp_2} \rceil$. Select $\alpha, a, b, t_{i,j} \in \mathbb{Z}_N$ randomly, and set $Y = X_1X_4 = g_1^a g_4^b$. Choose $\vec{\rho} \in \mathbb{Z}_N^\omega$ randomly and generate $PK = (N, g_1, g_4, \hat{e}(g_1, g_1)^\alpha, Y, g_1^{\vec{\rho}}, T_{i,j} = g_1^{t_{i,j}}; \forall i \in [n], j \in [n_i])$.

Phase 1: $\mathcal{B}$ generates normal keys for attribute sets in the key generation queries (keep in mind that the private keys of $\mathcal{A}$ query are either in normal form or in semi-functional form). In addition, $\mathcal{B}$ can answer the queries of leakage, reveal and update.

Challenge: $\mathcal{A}$ sends two equal length message M_0, M_1 and access structures Γ_0, Γ_1. $\mathcal{B}$ selects random $b \in \{0, 1\}$ and encrypts M_b under the access structure Γ_b. $\mathcal{B}$ encodes the access structure as the set of minimal sets $\mathcal{B}^* = \{B_1, B_2, ..., B_{\tilde{m}}\}$, where $B_k (k \in [\tilde{m}])$ is a set of attribute values. Select random element $s_1, s_2, ..., s_{\tilde{m}} \in Z_N$ and generate the challenge ciphertexts $C\bar{T}_\Gamma = (\{I_{B_k}\}_{k\in[\tilde{m}]}, C_0 = M_b\hat{e}(g_1^\alpha, T), \vec{C}_1 = T^{\vec{\rho}} * g_4^{\vec{d}}, C_2 = Tg_4, \vec{C}_3 = \{T^a(\prod_{v_{i,j}\in B_k} T_{i,j})^{s_k} W_k\}_{k\in[\tilde{m}]}, \vec{C}_4 = (g_1^{s_k} V_k)_{k\in[\tilde{m}]})$.

Phase 2: It is similar with Phase 1. $\mathcal{B}$ can answer the queries of reveal and update with the restriction that the attribute sets of adversary queries cannot meet the challenge access structure.

Guess: The adversary $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. If $b' = b$, $\mathcal{A}$ wins the game.

If $T \xleftarrow{\$} \mathbb{G}_{p_1p_2p_4}$, set T as g_2^c. It implicitly sets the semi-functional factor of the challenge ciphertexts as $(c\vec{\rho}, c, ac\vec{1}, 0)$. In this situation, $\mathcal{B}$ simulates the $Game_0$. Otherwise, $\mathcal{B}$ simulates the $Game_{real}$.

So if $\mathcal{A}$ can distinguish two games with a non-negligible advantage ϵ, $\mathcal{B}$ can use the algorithm to break the Assumption 1 with the same advantage.

Lemma 2. *Suppose there exists a PPT adversary $\mathcal{A}$ can distinguish $Game_{k'-1,2}$ and $Game_{k',1}$ with the non-negligible advantage ϵ, then is a PPT algorithm $\mathcal{B}$ with the advantage ϵ in breaking the Assumption 2.*

Proof $\mathcal{B}$ receives $E = (\mathbb{G}, g_1, g_3, g_4, U_1U_2, W_2W_3)$ and simulates $Game_{k'-1,2}$ or $Game_{k',1}$ with $\mathcal{A}$ depending on whether $T \xleftarrow{\$} \mathbb{G}_{p_1p_2p_3}$ or $T \xleftarrow{\$} \mathbb{G}_{p_1p_3}$.

Setup: It is same as Lemma 1.

Phase 1: Because $\mathcal{B}$ knows the master keys, it can answer all private key queries.

If $i' > k'$, $\mathcal{B}$ generates normal keys.

If $i' < k'$, $\mathcal{B}$ selects $t, h, y_2, y_3 \in \mathbb{Z}_N$ and $\vec{\sigma}, \vec{y}_1 \in \mathbb{Z}_N^\omega$ randomly, and picks $y_{i,j} \in \mathbb{Z}_N$ at random for generating Type 2 of semi-functional keys.

$$\begin{aligned} S\bar{K}_S &= (S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S) \\ &= (S, g_1^{\vec{\sigma}} * g_3^{\vec{y}_1}, g_1^{\alpha+\langle\vec{\rho},\vec{\sigma}\rangle} X_1^t (W_2W_3)^h g_3^{y_2}, g_1^t g_3^{y_3}, \\ &\quad T_{i,j}^t g_3^{y_{i,j}}; \forall v_{i,j} \in S) \end{aligned}$$

And if $i' = k'$, $\mathcal{B}$ selects $\vec{\sigma} \in \mathbb{Z}_N^\omega$ that satisfies $\langle\vec{\sigma}, \vec{\rho}\rangle = 0$, outputs the private key as follows:

$$\begin{aligned} S\bar{K}_S &= (S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S) \\ &= (S, T^{\vec{\sigma}} * g_3^{\vec{y}_1}, g_1^\alpha T^a g_3^{y_2}, Tg_3^{y_3}, T^{t_{i,j}} g_3^{y_{i,j}}; \\ &\quad \forall v_{i,j} \in S) \end{aligned}$$

If $T \xleftarrow{\$} \mathbb{G}_{p_1p_3}$, the private key is a normal key, $\mathcal{B}$ simulates the $Game_{k'-1,2}$. If $T \xleftarrow{\$} \mathbb{G}_{p_1p_2p_3}$, the private key is a semi-functional key of Type 1. In this case, $\mathcal{B}$ simulates $Game_{k',1}$.

Set the part of T in G_{p_2} is g_2^θ, then $\vec{d}_1 = \theta\vec{\sigma}, d_2 = a\theta, d_3 = \theta$. In addition, $\mathcal{A}$ can ask the oracles of leak and update.

Challenge: It is similar with Lemma 1. Set $U_1 = g_1^s, U_2 = g_2^\xi$ and calculate the ciphertexts

$$CT_\Gamma = (\{I_{B_k}\}_{k\in[\tilde{m}]}, C_0, \vec{C}_1, C_2, \vec{C}_3, \vec{C}_4),$$

in which

$$\begin{aligned} C_0 &= M_b\hat{e}(g_1^\alpha, U_1U_2), & \vec{C}_1 &= (U_1U_2)^{\vec{\rho}} * g_4^{\vec{d}}, \\ C_2 &= (U_1U_2)g_4, & \vec{C}_4 &= (g_1^{s_k} V_k)_{k\in[\tilde{m}]}, \\ \vec{C}_3 &= \{(U_1U_2)^a (\prod_{v_{i,j}\in B_k} T_{i,j})^{s_k} W_k\}_{k\in[\tilde{m}]}. \end{aligned}$$

Phase 2: It is similar with Phase 1. $\mathcal{B}$ can answer the queries of reveal and update with the restriction that the attribute sets corresponding to any private key of the query cannot satisfy the challenge access structure.

Guess: The adversary $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. If $b' = b$, $\mathcal{A}$ wins the game.

Based on the above descriptions, we obtain the semi-functional factor of the challenge ciphertexts is $(\xi\vec{\rho}, \xi, a\xi\vec{1}, 0)$. And the equation $\langle\vec{d}_1, \vec{e}_1\rangle - d_2e_2 + d_3e_{3,k} = 0$ holds. If the attributes of k'^{th} keys satisfy the challenge access structure, it is a nominally semi-functional key. Following the Lemma 5 in [30], the leakage of the key can not help adversary detect the k'^{th} private key is normal or semi-functional.

Lemma 3. *Suppose there exists a PPT adversary $\mathcal{A}$ can distinguish $Game_{k',1}$ and $Game_{k',2}$ with the non-negligible advantage ϵ, then there is a PPT algorithm $\mathcal{B}$ with the advantage ϵ in breaking the Assumption 2.*

Proof. Unlike construction of Lemma 2, the construction of k'^{th} key is as follows.

$$\begin{aligned} S\bar{K}_S =&(S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S) \\ =&(S, T^{\vec{\sigma}} * g_3^{\vec{y}_1}, g_1^{\alpha} T^a g_3^{y_2} (W_2 W_3)^d, T g_3^{y_3}, T^{t_{i,j}} g_3^{y_{i,j}}; \\ & \forall v_{i,j} \in S). \end{aligned}$$

□

If $T \xleftarrow{\$} \mathbb{G}_{p_1 p_3}$, this private key is a Type 2 of semi-functional key. Then $\mathcal{B}$ simulates the $Game_{k',2}$. If $T \xleftarrow{\$} \mathbb{G}_{p_1 p_2 p_3}$, this private key is the Type 1 of semi-functional key. In this case, $\mathcal{B}$ simulates $Game_{k',1}$. So if $\mathcal{A}$ can distinguish these two games with the advantage ϵ, $\mathcal{B}$ can break the Assumption 2 with the same advantage.

Lemma 4. *Suppose there is a PPT adversary $\mathcal{A}$ can distinguish $Game_{Q,2}$ and $Game_{final0}$ with the non-negligible advantage ϵ, then there exists a PPT algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 3.*

Proof. $\mathcal{B}$ receives the instance $E = (\mathbb{G}, g_1, g_2, g_3, g_4, g_2^r, U_2^r, g_1^{\alpha} U_2, g_1^s W_2)$, simulates $Game_{Q,2}$ or $Game_{final0}$. □

Setup: $\mathcal{B}$ selects $a, b, t_{i,j} \in \mathbb{Z}_N$, and sets $Y = X_1 X_4 = g_1^a g_4^b$. Select $\vec{\rho} \in \mathbb{Z}_N^{\omega}$ randomly and generate the $PK = (N, g_1, g_4, \hat{e}(g_1^{\alpha} U_2, g_1), Y, g_1^{\vec{\rho}}, T_{i,j} = g^{t_{i,j}}; \forall i \in [n], j \in [n_i])$.

Phase 1: All of the keys generated are the Type 2 of semi-functional keys. $\mathcal{B}$ selects $\vec{y}_1, \vec{\sigma} \in \mathbb{Z}_N^{\omega}, t, y_2, y_3 \in \mathbb{Z}_N$, $y_{i,j} \in \mathbb{Z}_N$ randomly, and outputs the secret keys $S\bar{K}_S = (S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S) = (S, g_1^{\vec{\sigma}} * g_3^{\vec{y}_1}, (g_1^{\alpha} U_2) X_1^t g_1^{\langle \vec{\rho}, \vec{\sigma} \rangle} g_3^{y_2}, g_1^t g_3^{y_3}, T_{i,j}^t g_3^{y_{i,j}}; \forall v_{i,j} \in S)$.

In addition, $\mathcal{A}$ can ask the oracles of leak and update.

Challenge: Similar to Lemma 1, $\mathcal{B}$ calculates the challenge ciphertexts $CT_{\Gamma} = (\{I_{B_k}\}_{k \in [\tilde{m}]}, C_0 = M_b T, \vec{C}_1 = (g_1^s U_2)^{\vec{\rho}} * g_4^{\vec{d}}, C_2 = (g_1^s U_2) g_4, \vec{C}_3 = \{(g_1^s U_2)^a \cdot (\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k} W_k\}_{k \in [\tilde{m}]}, \vec{C}_4 = (g_1^{s_k} V_k)_{k \in [\tilde{m}]})$.

Phase 2: It is similar with Phase 1. $\mathcal{B}$ can answer the queries of reveal and update with the restriction that the attribute sets corresponding to any private key of the query cannot satisfy the challenge access structure.

Guess: The adversary $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. If $b' = b$, $\mathcal{A}$ wins the game.

Obviously, we can learn that if $T = \hat{e}(g_1, g_1)^{\alpha s}$, it is a semi-functional ciphertext of message M_b. Otherwise $T \xleftarrow{\$} \mathbb{G}_T$, it is a random element of $\mathbb{G}_T$. So if the adversary $\mathcal{A}$ can distinguish these two games, $\mathcal{B}$ can break the assumption 3 with the same advantage.

Lemma 5. *Suppose there exists a PPT adversary $\mathcal{A}$ can distinguish $Game_{final0}$ and $Game_{final1}$ with the non-negligible advantage ϵ, then there is a PPT algorithm $\mathcal{B}$ with the advantage ϵ in breaking the Assumption 4.*

Proof. $\mathcal{B}$ receives the instance $E = (\mathbb{G}, g_1, g_2, g_3, g_4, U_1 U_4, U_1^{\hat{r}} U_2, g_1^{\hat{r}} W_2, g_1^s W_{24}, U_1 g_3^{\hat{s}})$ and simulates the $Game_{final0}$ or $Game_{final1}$. □

Setup: $\mathcal{B}$ selects $t_{i,j}, \alpha \in \mathbb{Z}_N$, $\vec{\rho} \in \mathbb{Z}_N^{\omega}$ randomly, and sets $PK = (N, g_1, g_4, \hat{e}(g_1, g_1)^{\alpha}, Y = U_1 U_4, g_1^{\vec{\rho}}, T_{i,j} = g^{t_{i,j}}; \forall i \in [n], j \in [n_i])$.

Phase 1: $\mathcal{B}$ selects $\vec{y}_1, \vec{\sigma} \in \mathbb{Z}_N^{\omega}, t, y_2, y_3 \in \mathbb{Z}_N, y_{i,j} \in \mathbb{Z}_N$ at random, calculates and outputs the secret keys as follows. $S\bar{K}_S = (S, \vec{K}_1, K_2, K_3, K_{i,j}; \forall v_{i,j} \in S) = (S, g_1^{\vec{\sigma}} * g_3^{\vec{y}_1}, g_1^{\alpha + \langle \vec{\rho}, \vec{\sigma} \rangle} (U_1 g_3^{\hat{s}})^t g_2 g_3^{y_2}, g_1^t g_3^{y_3}, T_{i,j}^t g_3^{y_{i,j}}; \forall v_{i,j} \in S)$.

In addition, $\mathcal{A}$ can ask the oracles of leak and update.

Challenge: $\mathcal{B}$ generates the challenge ciphertexts as follows: $C_0 \xleftarrow{\$} \mathbb{G}_T, \vec{C}_1 = (g_1^s W_{24})^{\vec{\rho}} * g_4^{\vec{d}}, C_2 = (g_1^s W_{24}) \cdot g_4, \vec{C}_3 = \{T(\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k} W_k\}_{k \in [\tilde{m}]}, \vec{C}_4 = (g_1^{s_k} V_k)_{k \in [\tilde{m}]}$.

Phase 2: It is similar with Phase 1. $\mathcal{B}$ can answer the queries of reveal and update with the restriction that the attributes sets of the adversary queries cannot meet the challenge access structure.

Guess: The adversary $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. If $b' = b$, $\mathcal{A}$ wins this game.

If $T \xleftarrow{\$} U_1^s D_{24}$, it is a semi-functional ciphertext, and $\mathcal{B}$ simulates the $Game_{final0}$. If $T \xleftarrow{\$} \mathbb{G}_{p_1 p_2 p_4}$, it is a random element, and $\mathcal{B}$ simulates the $Game_{final1}$. These two games are indistinguishable.

Theorem 1. *If the Assumptions 1, 2, 3 and 4 hold and for $l = (\omega - 1 - 2\tau)$ where τ is a positive constant, then the proposed scheme is anonymous and l leakage-resilient.*

Proof. Suppose the Assumption 1, 2, 3 and 4 hold. We can learn from the lemma 1 to lemma 5 that the adversary $\mathcal{A}$ can not distinguish between the $Game_{real}$ and $Game_{final1}$. So the value of b is hidden from the $\mathcal{A}$. The upper bound of the leakage between consecutive updates is l, so it is anonymous and l leakage-resilient. □

7 Anonymous Leakage-Resilient CP-ABE for Non-MAS

In this section, we give the construction of the anonymous leakage-resilient CP-ABE for non-monotone access structures. In this scheme, a non-monotone access structure is represented by the set of authorized sets in the non-monotone access structure.

Setup$((\lambda, V, l) \to (PK, MSK))$**:** Similar to Section 5, the setup algorithm sets the public keys as

$$PK = (N, g_1, g_4, g_1^{\vec{\rho}}, y, Y, T_{i,j}; \forall i \in [n], j \in [n_i]),$$

Table 2: Adversaries gain advantages over two consecutive games

Adjacent games	Adversary gain advantage differences	Related lemmas
$Game_{real}$ and $Game_0$	$\lvert Adv_{\mathcal{A}}^{Game_{real}} - Adv_{\mathcal{A}}^{Game_0}\rvert \leq \epsilon$	Lemma 1
$Game_{k'-1,2}$ and $Game_{k',1}$	$\lvert Adv_{\mathcal{A}}^{Game_{k'-1,2}} - Adv_{\mathcal{A}}^{Game_{k',1}}\rvert \leq \epsilon$	Lemma 2
$Game_{k',1}$ and $Game_{k',2}$	$\lvert Adv_{\mathcal{A}}^{Game_{k',1}} - Adv_{\mathcal{A}}^{Game_{k',2}}\rvert \leq \epsilon$	Lemma 3
$Game_{Q,2}$ and $Game_{final0}$	$\lvert Adv_{\mathcal{A}}^{Game_{Q,2}} - Adv_{\mathcal{A}}^{Game_{final0}}\rvert \leq \epsilon$	Lemma 4
$Game_{final0}$ and $Game_{final1}$	$\lvert Adv_{\mathcal{A}}^{Game_{final0}} - Adv_{\mathcal{A}}^{Game_{final1}}\rvert \leq \epsilon$	Lemma 5

where

$$y = e(g_1, g_1)^{\alpha}, Y = X_1X_4, T_{i,j} = g^{t_{i,j}}.$$

The master secret keys are

$$MSK = (X_1, g_3, \alpha).$$

Finally the algorithm publishs PK and keeps MSK.

KeyGen$((PK, MSK, S) \rightarrow SK_S)$**:** On input public keys PK, the master keys MSK, an attribute set $S = \{v_{1,x_1}, v_{2,x_2}, \cdots, v_{n',x'_n}\}$, where $n' \leq n, 1 \leq x_i \leq n_i$ for each $1 \leq i \leq n'$, this algorithm selects $t, y_2, y_3, y_4 \in \mathbb{Z}_N, \vec{y}_1, \vec{\sigma} \in \mathbb{Z}_N^{\omega}$, calculates and outputs the secret keys as follows.

$$SK_S = (S, \vec{K}_1, K_2, K_3, K_4),$$

in which

$$\vec{K}_1 = g_1^{\vec{\sigma}} * g_3^{\vec{y}_1}, \qquad K_3 = g_1^t g_3^{y_3},$$
$$K_2 = g_1^{\alpha + \langle \vec{\rho}, \vec{\sigma} \rangle} X_1^t g_3^{y_2}, \qquad K_4 = (\prod_{v_{i,j} \in S} T_{i,j})^t g_3^{y_4}.$$

UpdateUSK$((PK, S, SK_S) \rightarrow SK'_S)$**:** The update algorithm selects $\Delta t, \Delta y_2, \Delta y_3, \Delta y_4 \in \mathbb{Z}_N, \Delta\vec{y}_1, \Delta\vec{\sigma} \in \mathbb{Z}_N^{\omega}$ at random, and outputs a new key SK'_S:

$$SK'_S = (S, \vec{K}'_1, K'_2, K'_3, K'_4),$$

where

$$\vec{K}'_1 = \vec{K}_1 * g_1^{\Delta\vec{\sigma}} * g_3^{\Delta\vec{y}_1}, \quad K'_3 = K_3 g_1^{\Delta t} g_3^{\Delta y_3},$$
$$K'_2 = K_2 g_1^{\langle \vec{\rho}, \Delta\vec{\sigma} \rangle} X_1^{\Delta t} g_3^{\Delta y_2}, K'_4 = K_4 (\prod_{v_{i,j} \in S} T_{i,j})^{\Delta t} g_3^{\Delta y_4}.$$

Encrypt$((PK, M, \Gamma) \rightarrow CT_{\Gamma})$**:** Let Γ be a non-monotonic access structure where $\Gamma = \{B_1, B_2, ..., B_{\tilde{m}}\}$ and $B_k (k \in [\tilde{m}])$ is a set of attribute values and $\tilde{m}$ is the size of the non-monotone access structure Γ. This algorithm selects $s, s_1, s_2, ..., s_{\tilde{m}} \in \mathbb{Z}_N, \vec{d} \in \mathbb{Z}_N^{\omega}$, $W_k, V_k \in \mathbb{G}_{p_4} (k \in [\tilde{m}])$ at random and outputs the ciphertexts CT_{Γ} and the index $I_{B_k} \subset \{1, 2, ..., n\} (k \in [\tilde{m}])$ corresponding to attribute $B_k (k \in [\tilde{m}])$.

$$CT_{\Gamma} = (\{I_{B_k}\}_{k \in [\tilde{m}]}, C_0, \vec{C}_1, C_2, \vec{C}_3, \vec{C}_4),$$

where

$$C_0 = My^s, \qquad \vec{C}_1 = g_1^{s\vec{\rho}} * g_4^{\vec{d}},$$
$$C_2 = g_1^s g_4, \qquad \vec{C}_4 = (C_{4,k})_{k \in [\tilde{m}]} = (g_1^{s_k} V_k)_{k \in [\tilde{m}]},$$
$$\vec{C}_3 = \{C_{3,k}\}_{k \in [\tilde{m}]} = \{Y^s (\prod_{v_{i,j} \in B_k} T_{i,j})^{s_k} W_k\}_{k \in [\tilde{m}]}.$$

Decrypt$((PK, CT_{\Gamma}, SK_S) \rightarrow M)$**:** If the attribute set S satisfies the non-monotone access structure Γ, then $S \in \Gamma$, *i.e.*, $S = B_k (k \in [\tilde{m}])$ for $B_k \in \Gamma$. This algorithm calculates

$$M = \frac{C_0 \cdot \hat{e}_{\omega}(\vec{C}_1, \vec{K}_1) \hat{e}(C_{3,k}, K_3)}{\hat{e}(C_2, K_2) \hat{e}(C_{4,k}, K_4)}.$$

Theorem 2. *If Assumptions 1, 2, 3 and 4 hold, then the proposed scheme is anonymous and l leakage resilience.*

Proof. The security proof of anonymous leakage-resilient CP-ABE scheme for non-MAS can be derived from the proof of the scheme for MAS with minor modification. The minor modification is that in the simulation of key components for each $v_{i,j} \in S$ is just multiplied to get a single key component $K_4 = \prod_{v_{i,j} \in S} K_{i,j}$. □

8 Conclusion

In this paper, an anonymous leakage-resilient CP-ABE scheme for monotone access structures is proposed at first, in which the access structure is converted as minimal sets that can provide fast decryption. By using similar ideas, we present an anonymous leakage-resilient CP-ABE scheme with the constant size ciphertexts for non-monotone access structures. Both schemes are proven to be adaptively secure in the standard model under four static assumptions over composite order bilinear group and can tolerate continual leakage on the private keys when a update algorithm is implicitly employed to periodically update the private keys. However, our schemes cannot achieve the optimal leakage rate to ensure the efficiency, so designing an efficient ABE scheme with optimal leakage rate will be our future work.

Acknowledgments

This work was supported in part by the International S&T Cooperation Program of Shaanxi Province No. 2019KW-

056, the National Cryptography Development Fund under grant (MMJJ20180209).

References

[1] J. Alwen, Y. Dodis, M. Naor, G. Segev, S. Walfish, and D. Wichs, "Public-key encryption in the bounded-retrieval model," *Lecture Notes in Computer Science*, vol. 2009, no. 5, pp. 113–134, 2010.

[2] J. Alwen, Y. Dodis, and D. Wichs, "Leakage-resilient public-key cryptography in the bounded-retrieval model," in *Advances in Cryptology*, pp. 36–54, 2009.

[3] N. Attrapadug, B. Libert, and E. D. Panafieu, "Expressive key-policy attribute-based encryption with constant-size ciphertexts," in *Public Key Cryptography (PKC'11)*, pp. 90–108, 2011.

[4] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy*, pp. 321–334, 2007.

[5] Z. Cao, L. Liu, and Z. Guo, "Ruminations on attribute-based encryption," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 9–19, 2018.

[6] A. D. Caro, V. Iovino, and G. Persiano, "Fully secure anonymous hibe and secret-key anonymous ibe with short ciphertexts," in *International Conference on Pairing-Based Cryptography*, pp. 347–366, 2010.

[7] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments," *International Journal of Network Security*, vol. 16, no. 1, pp. 1–13, 2014.

[8] Y. Dodis, Y. T. Kalai, and S. Lovett, "On cryptography with auxiliary input," in *Proceedings of the 41st Annual ACM symposium on Theory of Computing*, pp. 621–630, 2009.

[9] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[10] A. Kapadia, P. Tsang, and S. W. Smith, "Attribute-based publishing with hidden credentials and hidden policies," *NDSS*, vol. 7, pp. 179–192, 2007.

[11] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Advances in Cryptology*, pp. 146–162, 2008.

[12] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *International Conference on Theory and Application of Cryptographic Techniques*, pp. 62–91, 2010.

[13] L. Liu, Z. Cao, and C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[14] S. Micali and L. Reyzin, "Physically observable cryptography," in *Theory of Cryptography*, pp. 278–296, 2004.

[15] T. Okamoto and K. Takashima, "Fully secure functional encyption with general relations from the decisional linear assumption," *Crypto*, vol. 6223, pp. 191–208, 2010.

[16] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access strtures," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS'07)*, pp. 195–203, 2007.

[17] T. Pandit and R. Barua, "Efficient fully secure attribute-based encryption schemes for general access structures," in *Provable Security*, pp. 193–214, 2012.

[18] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology*, pp. 457–473, 2005.

[19] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably seure realization," *Lecture Notes in Computer Science*, vol. 2008, pp. 321–334, 2008.

[20] B. Waters, "Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions," in *International Cryptology Conference on Advances in Cryptology*, pp. 619–636, 2009.

[21] Q. Yu and J. Li, "Continuous leakage resilient ciphertext-policy attribute-based encryption supporting attribute revocation," *Computer Engineering and Applications*, vol. 52, no. 20, pp. 29–38, 2016.

[22] T. H. Yuen, S. S. M. Chow, Y. Zhang, and S. M. Yiu, "Identity-based encryption resilient to continual auxiliary leakage," in *International Conference on Theory and Applications of Cryptographic Techniques*, pp. 117–134, 2012.

[23] L. Zhang, Y. Cui, and Y. Mu, "Improving privacy-preserving CP-ABE with hidden access policy," in *The 4th International Conference on Cloud Computing and Security*, pp. 596–605, 2018.

[24] L. Zhang, G. Hu, Y. Mu, and F. Rezaeibagha, "Hidden ciphertext policy attribute-based encryption with fast decryption for personal health record system," *IEEE Access*, no. 7, pp. 33202–33213, 2019.

[25] L. Zhang and Y. Shang, "Leakage-resilient attribute-based encryption with CCA2 security," *International Journal of Network Security*, vol. 22, no. 6, pp. 1–9, 2019.

[26] L. Zhang and H. Yin, "Recipient anonymous ciphertext-policy attribute-based broadcast encryption," *International Journal of Network Security*, vol. 20, no. 1, pp. 168–176, 2018.

[27] L. Zhang and J. Zhang, "Anonymous CP-ABE against side-channel attacks in cloud computing," *Journal of Information Science and Engineering*, vol. 33, no. 3, pp. 789–805, 2017.

[28] L. Zhang, J. Zhang, and Y. Hu, "Attribute-based encryption resilient to continual auxiliary leakage with

constant size ciphertexts," *Journal of China Universities of Posts and Telecommunications*, vol. 23, no. 3, pp. 18–28, 2016.

[29] L. Zhang, J. Zhang, and Y. Mu, "Novel leakage-resilient attribute-based encryption from hash proof system," *Computer Journal*, vol. 60, no. 4, pp. 541–554, 2017.

[30] M. Zhang, W. Shi, C. Wang, Z. Chen, and Y. Mu, "Leakage-resilient attribute-based encryption with fast decryption: Models, analysis and constructions," in *International Conference on Information Security Practice and Experience*, pp. 75–90, 2013.

Biography

Xiaoxu Gao is a master degree student in the school of mathematics and statistics, Xidian University. Her research interests focus on computer and network security.

Leyou Zhang is a professor in the school of mathematics and statistics at Xidian University, Xi'an China. He received his PhD from Xidian University in 2009. From Dec. 2013 to Dec. 2014, he is a research fellow in the school of computer science and software engineering at the University of Wollongong. His current research interests include network security, computer security, and cryptography.