JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 33, 789-805 (2017)
DOI: 10.6688/JISE.2017.33.3.12

Anonymous CP-ABE Against Side-Channel Attacks in Cloud Computing*

JING-XIA ZHANG[1] AND LE-YOU ZHANG[1,2]
[1]*Department of Mathematics and Statistics*
Xidian University
Shaanxi, 710126 P.R. China
[2]*Key Laboratory of Cryptography and Information Security in Guangxi*
Guangxi, 541004 P.R. China
E-mail: jingxiazhang92@163.com; lyzhang@mail.xidian.edu.cn

Anonymous ABE is a promising primitive for enforcing fine-grained access control for the big data as well as preserving privacy of the users in Cloud Computing. However, traditional anonymous ABE schemes may not be secure in the real word due to the side-channel attacks. In addition, the existing anonymous ABE schemes are considered in the leak-free scenario assuming that secret keys are not leaked to the adversary. Thus, it is compelling to study the anonymity of ABE schemes in the context of key leakage attacks. Aiming at tackling the challenge above, an anonymous CP-ABE scheme against side-channel attacks in the bounded-leakage model is constructed. As a main technique tool, the dual system encryption technique is adopted. The proposed scheme uses LSSS as access structures and achieves adaptive security in the standard model. In addition, the results in simulation experiments indicate that the proposed scheme is efficient and practical.

Keywords: anonymity, attribute-based encryption, leakage-resilient, fully secure, dual system encryption, side-channel attacks

1. INTRODUCTION

With the rapid development of cloud computing technology, more and more people have uploaded their various types of data into clouds, such as emails, personal health records, government documents, *etc.* By storing their data into the cloud, the data owners can be relieved from the burden of data storage and maintenance so as to enjoy the on-demand high quality applications and services. Especially for small and medium-sized enterprises with limited budgets, they can achieve cost savings and productivity enhancements by using cloud-based services to manage projects, to make collaborations, and the like. Naturally, people would like to make their private data only accessible to authorized users. To keep sensitive user data confidential against untrusted servers, a natural way is to apply cryptographic approaches, by disclosing decryption keys only to authorized users. In particular, the adopted encryption system should support fine-grained access control. That is, users with different attributes or roles should be granted different level of access privileges.

Attribute-based encryption (ABE) [1] is envisioned as a highly promising public

Received July 17, 2016; revised September 28, 2016; accepted October 22, 2016.
Communicated by Ram Chakka.
* This work was supported in part by the Nature Science Foundation of China under Grant (61472307, 61402-112, 61100165, 61100231), Natural Science Basic Research Plan in Shaanxi Province of China (Program No. 2016JM6004).

key primitive for realizing scalable and fine-grained access control enforced in cloud. Goyal *et al.* [2] formulated two forms of the ABE scheme. One is key-policy ABE (KP-ABE), and the other is ciphertext-policy ABE (CP-ABE). In KP-ABE, keys are associated with the access policy, and ciphertexts are associated with attribute sets. In CP-ABE, the situation is reversed: ciphertexts are associated with access policy, and keys are associated with the attribute sets. Although ABE supports fine-grained access control, there is an increasing need to protect user privacy in access control systems. Because many attributes are sensitive and related to the identity of eligible users. For example, the adversary can guess that the receiver is a student if some of the attributes are *school*, *class*, *score*, *etc.* Therefore, protecting receivers' identity while using ABE is a challenging research problem. In order to address this problem, anonymous ABE was introduced in references [3, 4] and further improved by references [5, 6]. In anonymous CP-ABE, access policy is hidden in the ciphertext. A user requires to decrypt a ciphertext using secret key belongs to his attributes. If the attributes of secret key satisfy the access policy, the user can successfully decrypt the ciphertext. If the attributes associated with the secret key do not satisfy the access policy, then the user cannot get what access policy is specified by the encryptor. Briefly speaking, anonymity requires that a ciphertext does not reveal information of its intended recipient.

Anonymous ABE has received a lot of attention in the past few years that mainly focus on providing security in the leak-free scenario assuming that the secret key is completely hidden from the adversary. Recent research has shown that various side-channel attacks, exploiting physical nature of cryptographic operations (*e.g.*, timing, power, radiation or cold-boot attacks), can extract some bits of information from secret keys and break the security of anonymous ABE schemes. That is, security is a major barrier to enterprise adoption of cloud computing. In practice, various side-channel attacks, exploiting physical nature of cryptographic operations (*e.g.*, timing, power, radiation or cold-boot attacks), can extract some bits of information from secret keys and break the security of anonymous ABE schemes. Responding to this challenge, it is natural to design cryptographic schemes which remain provably secure even in face of such attacks. Leakage-resilient cryptography was introduced to provide such security guarantees, which models a large class of side-channel attacks by allowing the adversary to specify an efficiently computable function f applied to secret keys in the security game. Leakage-resilience has been studied under a variety of leakage models. Some early models [7, 8] severely restricted the classes of allowed leakage available to the adversary. Recently, many excellent works [9-12] are proved secure in several different leakage models, such as the only computation leakage model and auxiliary input model. A simple and general leakage model, called bounded-leakage model, does not restrict the type of leakage that the adversary can obtain, but bound the overall amount of leakage by l bits. Here l is leakage bound of secret keys. Clearly the leakage bound l must be strictly smaller than the size of secret keys. Due to its elegance and generality, this model has attracted considerable attentions [13, 14].

1.1 Related Work

To achieve anonymity of receivers, Kapadia *et al.* [3] proposed a CP-ABE scheme, which can hide access policies that are represented by AND of different attributes, but it

is not collusion-resistant and needs an online semi-trusted server. Later, two efficient anonymous CP-ABE schemes are constructed in reference [5]. For the purpose of secure access control, Jin *et al.* [6] achieved user accountability, simultaneously still hide the receiver's attribute information in ciphertexts. However, in these anonymous ABE schemes, a user knows whether the attributes and the access policy match only after repeating decryption attempts, which lose practicability due to large computation cost. To resolve it, a novel technique called match-then-decrypt is proposed in reference [15], where a matching phase is additionally introduced before the decryption phase. It greatly improves the efficiency of decryption in anonymous ABE. Chaudhari *et al.* [16] further discussed the security weaknesses of the scheme [15] and proposed an improved scheme. Han *et al.* [17] gave an anonymous KP-ABE scheme, which achieves adaptive security based on three modified static assumptions in composite order bilinear groups.

For providing anonymity in the presence of side-channel attacks, Yu *et al.* [18] introduced the concept of anonymous identity-based hash proof system (IB-HPS), and showed how to construct leakage-resilient anonymous IBE schemes through anonymous IB-HPS in a generic way. Then, the scheme in reference [19] proposed an anonymous leakage-resilient identity-based broadcast encryption (IBBE) scheme. Recently, a continual leakage-resilient anonymous IBE scheme is proposed by Hu *et al.* [20]. However, all these work only focus on the security or anonymity. Thus, it is compelling to study the anonymity of ABE schemes in the context of side-channel attacks.

1.2 Our Contribution

In this work, we focus on how to construct an anonymous CP-ABE scheme secure against side-channel attacks. To this end, the dual system encryption technique is adopted, which is viewed as a powerful tool for achieving anonymity and leakage-resilience security. In composite order bilinear groups, the different subgroups of $\mathbb{G}$ play different roles in the cryptosystem. The normal encryption operation occurs in the subgroup $\mathbb{G}_{p_1}$, the subgroup $\mathbb{G}_{p_2}$ is a semi-functional space, and is only used in the security proof, the subgroup $\mathbb{G}_{p_3}$ is applied to randomize the secret keys, and the elements of the subgroup $\mathbb{G}_{p_4}$ are used to guarantee the anonymity of receivers. Based on the scheme [21], an anonymous CP-ABE scheme in the bounded-leakage model is proposed. The access structure used is LSSS, which is more flexible than AND gate. Under some static assumptions, the proposed scheme is fully secure. In addition, experimental results show that the proposed construction is efficient and practical.

1.3 Organization

The rest of this paper is organized as follows. Section 2 gives some preliminaries. Section 3 presents the definition and detailed construction of anonymous CP-ABE. The security analysis are presented in Section 4. The performance comparison in theoretically analysis and practical computation is presented in Section 5. Finally, we conclude this paper in Section 6.

2. PRELIMINARIES

In order to make the paper self-contained, we provide some preliminaries that have

been used throughout the paper.

2.1 Access Structures

Definition 1: [22] Let $\{P_1, P_2, \ldots, P_n\}$ be a set of parties. A collection $\mathbb{A} \subseteq 2^{\{P_1, P_2, \ldots, P_n\}}$ is monotonic if $\forall B \in \mathbb{A}$ and $B \subseteq C$, then $C \in \mathbb{A}$. An access structure is a collection $\mathbb{A}$ of non-empty subsets of $\{P_1, P_2, \ldots, P_n\}$. The sets in $\mathbb{A}$ are called the authorized sets, and the sets not in $\mathbb{A}$ are called the unauthorized sets.

In this context, attributes play the role of parties and we restrict our attention to monotonic access structures. It is possible to realize general access structures by treating the negation of an attribute as a separate attribute.

2.2 Linear Secret Sharing Schemes (LSSS)

Definition 2: [22] A secret sharing scheme $\prod$ over a set of parties $\mathcal{P}$ is called linear (over $\mathbb{Z}_p$) if

1. The shares for each party form a vector over $\mathbb{Z}_p$.
2. There exists a matrix A with n_1 rows and n_2 columns called the share-generating matrix for $\prod$. For all $i=1, 2, \ldots, n_1$, the ith row of A is labeled by a party $\rho(i)$, where ρ is a function from $\{1, 2, \ldots, n_1\}$ to $\mathcal{P}$. When we consider the column vector $\vec{v} = (s, r_2, \ldots, r_{n_2})$, where $s \in \mathbb{Z}_p$ is the secret to be shared, and $r_2, \ldots, r_{n_2} \in \mathbb{Z}_p$ are randomly chosen, then $A\vec{v}$ is the vector of n_1 shares of the secret s according to $\prod$. The share $(A\vec{v})_i$ belongs to party $\rho(i)$.

It is shown in [22] that every linear secret sharing scheme according to the above definition also enjoys the linear reconstruction property. Suppose that $\prod$ is an LSSS for the access structure $\mathbb{A}$. Let $S \in \mathbb{A}$ be any authorized set, and let $I \subset \{1, 2, \ldots, n_1\}$ be defined as $I = \{i: \rho(i) \in S\}$. Then, there exist constants $\{w_i \in \mathbb{Z}_p\}_{i \in I}$ such that, if $\{\lambda_i\}$ are valid shares of any secret s according to $\prod$, then $\sum_{i \in I} w_i \lambda_i = s$. These w_i can be found in time polynomial in the size of the share-generating matrix A.

2.3 Composite Order Bilinear Groups

Suppose that $\mathcal{G}$ is a group generator and λ is a security parameter. Composite order bilinear groups [23] can be defined as $(N = p_1 p_2 p_3 p_4, \mathbb{G}, \mathbb{G}_T, e) \leftarrow \mathcal{G}(1^\lambda)$, where p_1, p_2, p_3 and p_4 are four distinct primes, both $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of order N and the group operations in both $\mathbb{G}$ and $\mathbb{G}_T$ are computable in time polynomial in λ. A map $e: \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is an efficiently computable map with the following properties.

Bilinearity: $\forall a, b \in \mathbb{Z}_N$, and $g, h \in \mathbb{G}$, $e(g^a, h^b) = e(g, h)^{ab}$.
Non-degeneracy: $\exists g \in \mathbb{G}$ such that $e(g, g)$ has order N in $\mathbb{G}_T$.

Let $\mathbb{G}_{p_i p_j}$ denote the subgroup of order $p_i p_j$ for $i \neq j$ and $\mathbb{G}_{p_1}$, $\mathbb{G}_{p_2}$, $\mathbb{G}_{p_3}$ and $\mathbb{G}_{p_4}$ is defined as follows.

Definition 3: For all $u \in \mathbb{G}_{p_i}$, $v \in \mathbb{G}_{p_j}$, it holds that $e(u, v) = 1$ where $i \neq j$.

This orthogonality property can implement semi-functionality of the proposed system.

2.4 Complexity Assumptions

The following complexity assumptions are used to prove the security of the proposed schemes, which have been used in [20].

Assumption 1: Given a group generator $\mathcal{G}$, we define the following distribution:

$$(N = p_1p_2p_3p_4,\ \mathbb{G},\ \mathbb{G}_T,\ e) \xleftarrow{R} \mathcal{G}(1^\lambda),\ g_1 \xleftarrow{R} \mathbb{G}_{p_1},\ g_3 \xleftarrow{R} \mathbb{G}_{p_3},\ g_4 \xleftarrow{R} \mathbb{G}_{p_4},$$

$$D = (\mathbb{G},\ g_1,\ g_3,\ g_4),\ T_1 \xleftarrow{R} \mathbb{G}_{p_1p_4},\ T_2 \xleftarrow{R} \mathbb{G}_{p_1p_2p_4}.$$

The advantage of an algorithm $\mathcal{A}$ in breaking Assumption 1 is defined as

$$Adv1_{\mathcal{G},\mathcal{A}}(\lambda) = |\ Pr[\mathcal{A}(D,\ T_1) = 1] - Pr[\mathcal{A}(D,\ T_2) = 1]\ |.$$

Definition 4: We say that $\mathcal{G}$ satisfies Assumption 1 if $Adv1_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

Assumption 2: Given a group generator $\mathcal{G}$, we define the following distribution:

$$(N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, e) \xleftarrow{R} \mathcal{G}(1^\lambda)\ ,\ g_1, U_1 \xleftarrow{R} \mathbb{G}_{p_1}, U_2, Y_2 \xleftarrow{R} \mathbb{G}_{p_2}, g_3, Y_3 \xleftarrow{R} \mathbb{G}_{p_3}, g_4 \xleftarrow{R} \mathbb{G}_{p_4}$$

$$D = (\mathbb{G}, g_1, g_3, g_4, U_1U_2, Y_2Y_3), T_1 \xleftarrow{R} \mathbb{G}_{p_1p_2p_3}, T_2 \xleftarrow{R} \mathbb{G}_{p_1p_3}\ .$$

The advantage of an algorithm $\mathcal{A}$ in breaking Assumption 2 is defined as

$$Adv2_{\mathcal{G},\mathcal{A}}(\lambda) = |\ Pr[\mathcal{A}(D,\ T_1) = 1] - Pr[\mathcal{A}(D,\ T_2) = 1]\ |.$$

Definition 5: We say that $\mathcal{G}$ satisfies Assumption 2 if $Adv2_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

Assumption 3: Given a group generator $\mathcal{G}$, we define the following distribution:

$$(N = p_1p_2p_3p_4,\ \mathbb{G},\ \mathbb{G}_T,\ e) \xleftarrow{R} \mathcal{G}(1^\lambda)\ ,\ \alpha,\ s,\ r \xleftarrow{R} \mathbb{Z}_N,\ g_1 \xleftarrow{R} \mathbb{G}_{p_1},\ g_4 \xleftarrow{R} \mathbb{G}_{p_4}$$

$$U_2, Y_2, g_2 \xleftarrow{R} \mathbb{G}_{p_2}, g_3 \xleftarrow{R} \mathbb{G}_{p_3},$$

$$D = (\mathbb{G}, g_1, g_2, g_3, g_4, g_2^r, U_2^r, g_1^\alpha U_2, , g_1^s Y_2)\ ,\ \ T_1 = e(g_1, g_1)^{\alpha s}, T_2 \xleftarrow{R} \mathbb{G}_T\ .$$

The advantage of an algorithm $\mathcal{A}$ in breaking Assumption 3 is defined as

$$Adv3_{\mathcal{G},\mathcal{A}}(\lambda) = |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|.$$

Definition 6: We say that $\mathcal{G}$ satisfies Assumption 3 if $Adv3_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

Assumption 4: Given a group generator $\mathcal{G}$, we define the following distribution:

$$(N = p_1p_2p_3p_4,\ \mathbb{G},\ \mathbb{G}_T,\ e) \xleftarrow{R} \mathcal{G}(1^\lambda),\ \hat{r},\ \hat{s},\ s \xleftarrow{R} \mathbb{Z}_N,\ g_1,\ U_1 \xleftarrow{R} \mathbb{G}_{p_1},\ U_4,\ g_4 \xleftarrow{R} \mathbb{G}_{p_4},$$
$$g_2,\ U_2,\ B_2 \xleftarrow{R} \mathbb{G}_{p_2},\ g_3 \xleftarrow{R} \mathbb{G}_{p_3},\ B_{24},\ D_{24} \xleftarrow{R} \mathbb{G}_{p_2p_4},$$
$$D = (\mathbb{G},\ g_1,\ g_2,\ g_3,\ g_4,\ U_1U_4,\ U_1^{\hat{r}}U_2,\ g_1^{\hat{r}}B_2,\ g_1^{s}B_{24},\ U_1g_3^{\hat{s}}),\ T_1 \xleftarrow{R} U_1^{s}D_{24},\ T_2 \xleftarrow{R} \mathbb{G}_{p_1p_2p_4}.$$

The advantage of an algorithm $\mathcal{A}$ in breaking Assumption 4 is defined as

$$Adv4_{\mathcal{G},\mathcal{A}}(\lambda) = |\ Pr[\mathcal{A}(D,\ T_1) = 1] - Pr[\mathcal{A}(D,\ T_2) = 1]\ |.$$

Definition 7: We say that $\mathcal{G}$ satisfies Assumption 4 if $Adv4_{\mathcal{G},\mathcal{A}}(\lambda)$ is a negligible function of λ for any polynomial time algorithm $\mathcal{A}$.

3. ANONYMOUS CP-ABE

3.1 Definition of CP-ABE

A CP-ABE scheme consists of the following four algorithms.

Setup(1^λ, U)$\rightarrow$(PK, MSK): This algorithm takes as input a security parameter λ and attribute universe description U. It then outputs the public key PK and the master secret key MSK.

KeyGen(MSK, PK, S)$\rightarrow SK$: For any a set of attributes S, this algorithm uses the master secret key MSK and public key PK to sample a secret key SK.

Encrypt(PK, M, $\mathbb{A}$)$\rightarrow CT$: This algorithm takes in the public key PK, the message M, and the access structure $\mathbb{A}$, and then it outputs the ciphertext CT.

Decrypt(PK, CT, SK)$\rightarrow M$: The decryption algorithm takes as input the public key PK, a ciphertext CT and secret key SK. If the attribute sets of the secret key satisfy the access structure $\mathbb{A}$, it outputs the message M.

3.2 Detailed Construction

Setup(λ, U): First, the algorithm chooses a bilinear group $\mathbb{G}$ of order $N=p_1p_2p_3p_4$. The subgroup of order p_i(i=1, 2, 3, 4) in $\mathbb{G}$ is denoted by $\mathbb{G}_{p_i}$. n is a positive integer greater than or equal to 2. Higher values of n will lead to a better fraction of leakage being tolerated, while lower values of n will yield a system with fewer group elements in the keys and ciphertexts. It then picks random values $\alpha \in \mathbb{Z}_N$, g_1, $X_1 \in \mathbb{G}_{p_1}$, $g_3 \in \mathbb{G}_{p_3}$, g_4, $X_4 \in \mathbb{G}_{p_4}$ and

sets $Y=X_1X_4$ For each attribute $i\in U$, the algorithm chooses random $s_i\in\mathbb{Z}_N$. It also picks random exponents $x_1, \ldots, x_n\in \mathbb{Z}_N$, to get the required vectors. The public key PK and master secret key MSK are

$$PK=\{N, g_1, g_4, e(g_1, g_1)^{\alpha}, Y, g_1^{x_1}, \ldots, g_1^{x_n}, T_i=g_1^{s_i}, \forall i\in U\}. \text{ MSK}=\{X_1, g_3, \alpha\}.$$

KeyGen(*PK*, *MSK*, *S*): It chooses random $t, z_1, \ldots, z_n\in\mathbb{Z}_N$, $\vec{\eta}\in\mathbb{Z}_N^{n+1}$ and $R, R_i\in\mathbb{G}_{p_3}$ for each attribute $i\in S$. The secret key SK is given as

$$\overrightarrow{K_1}=< g_1^{z_1}, \cdots, g_1^{z_n}, g_1^{\alpha}X_1^t\prod_{i=1}^{n}g_1^{-x_iz_i} > *g_3^{\vec{\eta}}, L=g_1^tR, K_i=T_i^tR_i\forall i\in S.$$

Encrypt(*PK*, *M*, (*A*, ρ)): A is an $n_1\times n_2$ matrix and ρ is a map from each row A_x of A to an attribute $\rho(x)\in U$. It chooses a random vector $\vec{v}=(s, v_2, \cdots, v_{n_2})\in\mathbb{Z}_N^{n_2}$ and $\vec{d}\in\mathbb{Z}_N^{n+1}$. For each row A_x of A, it picks random values $r_x\in\mathbb{Z}_N$, and $W_x, V_x\in\mathbb{G}_{p_4}$.The ciphertext CT is given as

$$C=Me(g_1, g_1)^{\alpha s}, \overrightarrow{C_1}=< g_1^{x_1s}, \cdots, g_1^{x_ns}, g_1^s > *g_4^{\vec{d}}, C_x=Y^{A_x\vec{v}}T_{\rho(x)}^{-r_x}W_x, D_x=g_1^{r_x}V_x.$$

Decrypt(*CT*, *PK*, *SK*): If S satisfies the access structure (A, ρ), the algorithm computes $w_x\in\mathbb{Z}_N$ such that $\sum_{\rho(x)\in S}w_xA_x=(1, 0, \cdots, 0)$. Then the algorithm calculates

$$\frac{e_{n+1}(\overrightarrow{C_1}, \overrightarrow{K_1})}{\prod_{\rho(x)\in S}(e(C_x, L)e(D_x, K_{\rho(x)}))^{w_x}}=\frac{e(g_1^{x_1s}, g_1^{z_1})e(g_1^{x_2s}, g_1^{z_2})\cdots e(g_1^{x_ns}, g_1^{z_n})e(g_1^s, g_1^{\alpha}X_1^t\prod_{i=1}^{n}g_1^{-x_iz_i})}{\prod_{\rho(x)\in S}e(g_1^t, Y^{A_x\vec{v}})^{w_x}e(g_1^t, T_{\rho(x)}^{-r_x})e(g_1^{r_x}, T_{\rho(x)}^t)}$$

$$=\frac{e(g_1^s, g_1^{\alpha}X_1^t)}{e(g_1^t, Y^s)}=\frac{e(g_1^s, g_1^{\alpha}X_1^t)}{e(g_1^t, X_1^s)}$$

$$=e(g_1, g_1)^{\alpha s},$$

and the message can be recovered by $C/e(g_1, g_1)^{\alpha s}=Me(g_1, g_1)^{\alpha s}/e(g_1, g_1)^{\alpha s}=M$.

4. SECURITY ANALYSIS

4.1 Security Model

We demonstrate security requirements for anonymous CP-ABE systems in the presence of side-channel attacks by modeling the capability of adversaries, and define corresponding security notions.

Confidentiality: This is the usual security notion of semantic security for encryption. It means that in face of key leakage attacks, no non-trivial information about the message can be feasibly gleaned from the ciphertext.

Anonymity: Recipient anonymity is the property that the adversary be unable to distinguish the encryption of a chosen message for a first chosen access policy from the en-

cryption of the same message for a second chosen access policy.

Equivalently, the goals of an adversary in an anonymous CP-ABE system include extracting information of a plaintext from the ciphertext in presence of side-channel attacks and distinguishing underlying access policies in ciphertexts, which can be integrated the following game (against chosen-plaintext attacks) in the bounded-leakage model. The security game is parameterized by a security parameter λ and leakage bound l.

Setup: The challenger $\mathcal{B}$ runs $(PK, MSK) \leftarrow Setup(1^{\lambda}, U)$ and gives PK to the adversary $\mathcal{A}$. $\mathcal{B}$ will construct a set $\mathcal{Q}$ of tuple of handles, attributes sets, secret keys it has created and the number of leaked bits, that is, (h, S, SK, L_{SK}), but does not revealed it. Also $\mathcal{B}$ will construct a set $\mathcal{R}$ of attribute sets whose secret keys have been revealed.

Phase 1: $\mathcal{A}$ adaptively makes the following queries.

Create Queries: $\mathcal{A}$ gives attribute sets S to the challenger. $\mathcal{B}$ creates SK, sets $h=h+1$ and adds $(h, S, SK, 0)$ to the set $\mathcal{Q}$. In this query, $\mathcal{B}$ only gives the adversary a handle h rather than the concrete secret key itself.

Leakage Queries: $\mathcal{A}$ gives a PPT function $f: \{0, 1\}^* \rightarrow \{0, 1\}^*$ with a queried handle h of key to the challenger. The challenger replies with $f(SK)$ if $L_{SK}+|f(SK)| \leq l$ and updates L_{SK} with $L_{SK}+|f(SK)|$. Otherwise, outputs $\perp$.

Reveal Queries: $\mathcal{A}$ gives the handle for a specified key SK to the challenger. $\mathcal{B}$ scans $\mathcal{Q}$ to find the requested entry and returns the secret key to $\mathcal{A}$. Then the challenger removes the item from the set $\mathcal{Q}$ and adds the attribute sets into the set $\mathcal{R}$.

Challenge: $\mathcal{A}$ outputs two pairs of equal length messages and access structures $(M_0, \mathbb{A}_0)$, $(M_1, \mathbb{A}_1)$ to the challenger where every attribute sets $S \in \mathcal{R}$ does not satisfy $\mathbb{A}_0$ and $\mathbb{A}_1$. $\mathcal{B}$ randomly chooses $\beta \in \{0, 1\}$ and encrypts M_β with $\mathbb{A}_\beta$.

Phase 2: This phase is the same as Phase 1 with the restriction that only *Reveal Queries* involving secret keys whose attribute sets does not satisfy the challenge access structure can be queried.

Guess: The adversary outputs a guess $\beta' \in \{0, 1\}$.

Definition 8: An anonymous CP-ABE scheme is l-leakage-resilient secure if the advantage of any PPT adversary $\mathcal{A}$ in the previously mentioned game is negligible, where the advantage of $\mathcal{A}$ is defined as $Adv_{\mathcal{A}}(\lambda, l) = |Pr[\beta' = \beta] - \frac{1}{2}|$.

4.2 Security Proof for Anonymous CP-ABE

Two additional structures are defined in this section. These will not be used in the real scheme, but we need them in our proofs.

Semi-functional key: There are two types of semi-functional key. A semi-functional key

of type 1 is formed as follows. The algorithm picks $\vec{\gamma} \in \mathbb{Z}_N^{n+1}$, $\theta \in \mathbb{Z}_N$, and $q_i \in \mathbb{Z}_N$ for each attribute $i \in S$. It outputs

$$\overrightarrow{K_1} = < g_1^{z_1}, \cdots, g_1^{z_n}, g_1^{\alpha} X_1^t \prod_{i=1}^{n} g_1^{-x_i z_i} > * g_3^{\vec{\eta}} * g_2^{\vec{\gamma}}, \; L = g_1^t R g_2^{\theta}, \; K_i = T_i^t R_i g_2^{\theta q_i}.$$

A semi-functional key of type 2 is formed without the terms g_2^{θ} and $g_2^{\theta q_i}$. The key is set as

$$\overrightarrow{K_1} = < g_1^{z_1}, \cdots, g_1^{z_n}, g_1^{\alpha} X_1^t \prod_{i=1}^{n} g_1^{-x_i z_i} > * g_3^{\vec{\eta}} * g_2^{\vec{\gamma}}, \; L = g_1^t R, \; K_i = T_i^t R_i.$$

Semi-functional ciphertext: A semi-functional ciphertext is formed as follows. The algorithm picks $\vec{\delta} \in \mathbb{Z}_N^{n+1}$ and a random vector $\vec{u} \in \mathbb{Z}_N^{n_2}$ (n_2 is the number of columns of A). For every row A_x of A, it chooses $h_x \in \mathbb{Z}_N$ and outputs

$$C = Me(g_1, g_1)^{\alpha s}, \; \overrightarrow{C_1} = < g_1^{x_1 s}, \ldots, g_1^{x_n s}, g_1^{s} > * g_4^{\vec{d}} * g_2^{\vec{\delta}}, \quad C_x = Y^{A_x \vec{v}} T_{\rho(x)}^{-r_x} W_x g_2^{A_x \vec{u} + h_x q_{\rho(x)}},$$
$$D_x = g_1^{r_x} V_x g_2^{-h_x}.$$

When a semi-functional key decrypts a semi-functional ciphertext, the extra term $e(g_2, g_2)^{\vec{\gamma}\vec{\delta} - u_1 \theta}$ arises where u_1 denotes the first coordinate of $\vec{u}$. A semi-functional key of type 1 is called nominally semi-functional if $\vec{\gamma}\vec{\delta} - u_1 \theta \equiv 0 mod p_2$. When such a key decrypts a corresponding semi-functional ciphertext, decryption still works.

For a probabilistic polynomial-time adversary $\mathcal{A}$ which makes q key queries, our proof of security will consist of the following sequence of games between $\mathcal{A}$ and a challenger $\mathcal{B}$.

$Game_{real}$: This is the real security game (the challenge ciphertext and all keys are normal).

$Game_0$: This is the same as the real security game except that the challenge ciphertext is semi-functional.

$Game_{k,1}$: For k from 1 to q, $Game_{k,1}$ like $Game_0$ except that the first k-1 keys are semi-functional of type 2, the kth key is semi-functional of type 1.

$Game_{k,2}$: In this game, the challenge ciphertext is semi-functional. The first k keys are semi-functional of type 2, and the remaining keys are normal.

$Game_{final0}$: This game is the same as $Game_{q,2}$, except that the challenge ciphertext is a semi-functional encryption with C random in $\mathbb{G}_T$. Thus the ciphertext is independent from the messages provided by $\mathcal{A}$.

$Game_{final1}$: This game is the same as $Game_{final0}$, except that the challenge ciphertext is a semi-functional encryption with C_x random in $\mathbb{G}_{p_1 p_2 p_4}$. Obviously in $Game_{final1}$, the ciphertext is independent from the access structures provided by $\mathcal{A}$ and the adversary's advantage is 0. The indistinguishability of these games are proved by the following lemmas.

Lemma 1: Suppose there is a PPT algorithm $\mathcal{A}$ such that $Adv_{\mathcal{A}}(Game_{real})-Adv_{\mathcal{A}}(Game_0)=\varepsilon$, we can build a PPT algorithm $\mathcal{B}$ with advantage at least ε in breaking Assumption 1.

***Proof*:** Given $(\mathbb{G}, g_1, g_3, g_4)$ and T from Assumption 1, $\mathcal{B}$ simulates $Game_{real}$ or $Game_0$ with the adversary depending on whether $T \xleftarrow{R} \mathbb{G}_{p_1 p_4}$ or $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_4}$.

Setup: $\mathcal{B}$ picks random exponents a, b, $\alpha \in \mathbb{Z}_N$ and $s_i \in \mathbb{Z}_N$ for each attribute I in the system. It implicitly sets $Y=X_1X_4=g_1^a g_4^b$ and picks random exponents $x_1, \ldots, x_n \in \mathbb{Z}_N$. The public key is given as

$$PK = \{N, g_1, g_4, e(g_1, g_1)^{\alpha}, Y, g_1^{x_1}, \cdots, g_1^{x_n}, T_i = g_1^{s_i}, \forall i \in U\}.$$

Phase 1: Knowing the master secret key, $\mathcal{B}$ can generate normal secret keys in response to all secret key requests (*create*, *leakage*, *reveal queries*).

Challenge: The adversary sends $\mathcal{B}$ two equal length messages M_0, M_1 and access structures $\mathbb{A}_0$, $\mathbb{A}_1$. $\mathcal{B}$ randomly chooses $\beta \in \{0, 1\}$ and encrypts M_β with $\mathbb{A}_\beta$. The access structure $\mathbb{A}_\beta$ is encoded as an $n_1 \times n_2$ LSSS matrix (A^*, ρ^*). To make the challenge ciphertext, $\mathcal{B}$ implicitly sets g_1^s to be the $\mathbb{G}_{p_1}$ part of T and chooses random values $v_2', \ldots, v_{n_2}', r_x' \in \mathbb{Z}_N$. It then creates the vector $\vec{v}' = (1, v_2', \cdots, v_{n_2}')$ and sets

$$C = M_\beta e(g_1^{\alpha}, T),\ \overrightarrow{C_1} = < T^{x_1}, \cdots, T^{x_n}, T > * g_4^{\vec{d}},\ C_x = T^{aA_x^*\vec{v}'} T^{-s_{\rho^*(x)} r_x'} W_x,\ D_x = T^{r_x'} V_x.$$

Phase 2: $\mathcal{B}$ works in the same way as Phase 1.

Thus if $T \xleftarrow{R} \mathbb{G}_{p_1 p_4}$, this implicitly sets $\vec{v} = s\vec{v}', r_x = sr_x'$. This is a correctly distributed normal ciphertext and $\mathcal{B}$ has properly simulated $Game_{real}$. If $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_4}$, let g_2^{ξ} denote the $\mathbb{G}_{p_2}$ part of T. Then the above is a semi-function ciphertext with $\vec{u} = \xi a\vec{v}'$, $h_x = -\xi r'$, $\vec{\delta} = \xi < x_1, \ldots, x_n, 1>$ and $q_{\rho^*(x)} = s_{\rho^*(x)}$. The values of $a, v_2', \cdots, v_{n_2}', r_x', s_{\rho(x)}, x_1, \cdots, x_n$ modulo p_2 are uncorrelated with their values modulo p_1, so this is a properly distributed semi-functional ciphertext and $\mathcal{B}$ has properly simulated $Game_0$. Hence, $\mathcal{B}$ can use the output of $\mathcal{A}$ to gain advantage ε in breaking Assumption 1.

Lemma 2: Suppose there is a PPT algorithm $\mathcal{A}$ such that $Adv_{\mathcal{A}}(Game_{k\text{-}1,2}) - Adv_{\mathcal{A}}(Game_{k,1}) = \varepsilon$, we can build a PPT algorithm $\mathcal{B}$ with advantage at least ε in breaking Assumption 2.

***Proof*:** Given $(\mathbb{G}, g_1, g_3, g_4, U_1U_2, Y_2Y_3)$ and T from Assumption 2, $\mathcal{B}$ simulates $Game_{k\text{-}1,2}$ or $Game_{k,1}$ with the adversary depending on whether $T \xleftarrow{R} \mathbb{G}_{p_1 p_3}$ or $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_3}$.

Setup: $\mathcal{B}$ generates the public key *PK* in the same way as Lemma 1.

Phase 1: Since $\mathcal{B}$ knows *MSK*, it can answer all secret key queries for normal keys or semi-functional keys. $\mathcal{B}$ can form normal keys for queries $> k$ via the regular key generation algorithm. To create semi-functional keys of type 2 for queries $< k$, $\mathcal{B}$ randomly chooses $t, z_1, \cdots, z_n \in \mathbb{Z}_N, \vec{\psi} \in \mathbb{Z}_N^{n+1}$ and R, $R_i \in \mathbb{G}_{p_3}$. The semi-functional keys of type 2 can be defined as

$$\overrightarrow{K_1} =< g_1^{z_1}, \cdots, g_1^{z_n}, g_1^{\alpha} X_1^t \prod_{i=1}^{n} g_1^{-x_i z_i} > *(Y_2 Y_3)^{\vec{\psi}}, L = g_1^t R, K_i = T_i^t R_i.$$

For the *k*th key, $\mathcal{B}$ has to either give a normal key or a semi-functional key of type 1. To do this, it picks $z'_1, \ldots, z'_n \in \mathbb{Z}_N$ and implicitly sets g_1^t equal to the $\mathbb{G}_{p_1}$ part of T. The secret key is given as

$$\overrightarrow{K_1} =< T^{z'_1}, \cdots, T^{z'_n}, g_1^{\alpha} T^a \prod_{i=1}^{n} T^{-x_i z'_i} > *g_3^{\vec{\eta}}, L = TR, K_i = T^{s_i} R_i.$$

If $T \xleftarrow{R} \mathbb{G}_{p_1 p_3}$, this is a properly distributed normal key. If $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_3}$, this is a semi-functional key of type 1. In this case, it implicitly sets g_2^{θ} equal to the $\mathbb{G}_{p_2}$ part of T and then $\vec{\gamma} = \theta < z'_1, \cdots, z'_n, a - x_i z'_i >, q_i = s_i$.

Challenge: The adversary $\mathcal{A}$ sends $(M_0, \mathbb{A}_0)$ and $(M_1, \mathbb{A}_1)$ to $\mathcal{B}$. It randomly chooses $\beta \in \{0, 1\}$ and encrypts M_β with $\mathbb{A}_\beta$. The access structure $\mathbb{A}_\beta$ is encoded as an $n_1 \times n_2$ LSSS matrix (A^*, ρ^*). To form the challenge ciphertext, $\mathcal{B}$ implicitly sets $U_1 = g_1^s$, $U_2 = g_2^{\xi}$ and chooses random exponent $r'_x \in \mathbb{Z}_N$. It also picks random values $u_2, \cdots, u_{n_2} \in \mathbb{Z}_N$ and defines the vector $\vec{u}'$ as $\vec{u}' = (a, u_2, \cdots, u_{n_2})$. $\mathcal{B}$ sets

$$C = M_\beta e(g_1^{\alpha}, U_1 U_2), \overrightarrow{C_1} =< (U_1 U_2)^{x_1}, \cdots, (U_1 U_2)^{x_n}, U_1 U_2 > *g_4^{\vec{d}},$$

$$C_x = (U_1 U_2)^{A_x^* \vec{u}'} (U_1 U_2)^{-s_{\rho^*(x)} r'_x} W_x, D_x = (U_1 U_2)^{r'_x} V_x.$$

For the $\mathbb{G}_{p_1}$ parts, this implicitly sets $\vec{v} = sa^{-1}\vec{u}'$ and $r_x = sr'_x$. For the $\mathbb{G}_{p_2}$ parts, it sets $\vec{u} = \xi \vec{u}', h_x = -\xi r'_x, \vec{\delta} = \xi < x_1, \cdots, x_n, 1 >$ and $q_{\rho(x)} = s_{\rho(x)}$. From the previous procedure of creating the *k*th key and the semi-functional challenge ciphertext, we can see that if the *k*th key is semi-functional and the attributes of this key satisfy the challenge access structure, it is nominally semi-functional since $\vec{\gamma}\vec{\delta} - \theta u_1 \equiv 0 \bmod p_2$. According to the rules of the game, this key cannot be revealed to the adversary, but only leakage is allowed on it. By Corollary 6.3 in [21], the leakage of the secret key cannot help the adversary check whether the *k*th key is normal or semi-functional.

Phase 2: $\mathcal{B}$ works in the same way as Phase 1.

Thus if $T \xleftarrow{R} \mathbb{G}_{p_1 p_3}$, $\mathcal{B}$ simulates $Game_{k\text{-}1,2}$. If $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_3}$, $\mathcal{B}$ simulates $Game_{k,1}$. Hence, $\mathcal{B}$ can use the output of $\mathcal{A}$ to gain advantage ε in breaking Assumption 2.

Lemma 3: Suppose there is a PPT algorithm $\mathcal{A}$ such that $Adv_{\mathcal{A}}(Game_{k,1}) - Adv_{\mathcal{A}}(Game_{k,2}) = \varepsilon$, we can build a PPT algorithm $\mathcal{B}$ with advantage at least ε in breaking Assumption 2.

***Proof*:** This proof is very similar to the proof of Lemma 2 except that the *k*th key is different from it. To make the *k*th secret key, it additionally chooses a random vector $\vec{\psi} \in \mathbb{Z}_N^{n+1}$ and sets

$$\overrightarrow{K_1} =< T^{z'_1}, \cdots, T^{z'_n}, g_1^{\alpha} T^a \prod_{i=1}^{n} T^{-x_i z'_i} > *g_3^{\vec{\eta}} * (Y_2 Y_3)^{\vec{\psi}}, L = TR, K_i = T^{s_i} R_i.$$

Obviously, if $T \xleftarrow{R} \mathbb{G}_{p_1 p_3}$, this is a properly distributed semi-functional key of type 2 and $\mathcal{B}$

simulates $Game_{k,2}$. If $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_3}$, this is a properly distributed semi-functional key of type 1 and $\mathcal{B}$ simulates $Game_{k,1}$. Hence, $\mathcal{B}$ can use the output of $\mathcal{A}$ to gain advantage ε in breaking Assumption 2.

Lemma 4: Suppose there is a PPT algorithm $\mathcal{A}$ such that $Adv_{\mathcal{A}}(Game_{q,2}) - Adv_{\mathcal{A}}(Game_{final0}) = \varepsilon$, we can build a PPT algorithm $\mathcal{B}$ with advantage at least ε in breaking Assumption 3.

***Proof*:** Given $(\mathbb{G}, g_1, g_2, g_3, g_4, g_2^r, U_2^r, g_1^\alpha U_2, g_1^s Y_2)$ and T from Assumption 3, $\mathcal{B}$ simulates $Game_{q,2}$ or $Game_{final0}$ with the adversary depending on whether $T = e(g_1, g_1)^{\alpha s}$ or $T \xleftarrow{R} \mathbb{G}_T$.

Setup: The adversary picks random exponents $a, b, x_1, \cdots, x_n, s_i \in \mathbb{Z}_N$ and takes α from the assumption term $g_1^\alpha X_2$. It sets $Y = X_1 X_4 = g_1^a g_4^b$ and sends $\mathcal{A}$ the public key

$$PK = \{N, g_1, g_4, e(g_1^\alpha U_2, g_1), Y, g_1^{x_1}, \cdots, g_1^{x_n}, T_i = g_1^{s_i}, \forall i \in U\}.$$

Phase 1: All keys generated by $\mathcal{B}$ should be semi-functional keys of type 2. The challenger picks random $t, z_1, \cdots, z_n \in \mathbb{Z}_N, \vec{\eta}, \vec{\gamma} \in \mathbb{Z}_N^{n+1}$ and $R, R_i \in \mathbb{G}_{p_3}$ for each attribute $i \in S$. It creates the following secret key

$$\overrightarrow{K_1} = < g_1^{z_1}, \cdots, g_1^{z_n}, g_1^\alpha U_2 X_1^t \prod_{i=1}^{n} g_1^{-x_i z_i} > * g_2^{\vec{\gamma}} * g_3^{\vec{\eta}}, L = g_1^t R, K_i = T_i^t R_i.$$

Challenge: To form the challenge ciphertext, $\mathcal{B}$ chooses random values $v_2', \cdots, v_{n_2}', r_x' \in \mathbb{Z}_N$. It then creates the vector $\vec{v}' = (1, v_2', \cdots, v_{n_2}')$ and sets

$$C = M_\beta T, \overrightarrow{C_1} = < (g_1^s Y_2)^{x_1}, \cdots, (g_1^s Y_2)^{x_n}, g_1^s Y_2 > * g_4^{\vec{d}}, \; C_x = (g_1^s Y_2)^{\alpha A_x^* \vec{v}'} (g_1^s Y_2)^{-s_{\rho^*(x)} r_x'} W_x, D_x = (g_1^s Y_2)^{r_x'} V_x.$$

Phase 2: $\mathcal{B}$ works in the same way as Phase 1.

It is obvious that if $T = e(g_1, g_1)^{\alpha s}$, the above is a properly distributed semi-functional encryption of M_β and $\mathcal{B}$ simulates $Game_{q,2}$. Otherwise, it is an encryption of a random message and $\mathcal{B}$ simulates $Game_{final0}$. Hence, $\mathcal{B}$ can use the output of $\mathcal{A}$ to gain advantage ε in breaking Assumption 3.

Lemma 5: Suppose there is a PPT algorithm $\mathcal{A}$ such that $Adv_{\mathcal{A}}(Game_{final0}) - Adv_{\mathcal{A}}(Game_{final1}) = \varepsilon$, we can build a PPT algorithm $\mathcal{B}$ with advantage at least ε in breaking Assumption 4.

***Proof*:** Given $(\mathbb{G}, g_1, g_2, g_3, g_4, U_1 U_4, U_1^{\hat{r}} U_2, g_1^{\hat{r}} B_2, g_1^s B_{24}, U_1 g_3^{\hat{s}})$ and T from Assumption 4, $\mathcal{B}$ simulates $Game_{final0}$ or $Game_{final1}$ with the adversary depending on whether $T \xleftarrow{R} U_1^s D_{24}$ or $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_4}$.

Setup: The adversary picks random exponents $\alpha, x_1, \ldots, x_n, s_i \in \mathbb{Z}_N$ and sets $Y = U_1 U_4$. $\mathcal{B}$ sends $\mathcal{A}$ the public key

$$PK = \{N, g_1, g_4, e(g_1, g_1)^\alpha, Y, g_1^{x_1}, \cdots, g_1^{x_n}, T_i = g_1^{s_i}, \forall i \in U\}.$$

Phase 1: All keys generated by $\mathcal{B}$ should be semi-functional keys of type 2. The challenger picks random $t, z_1', \cdots, z_n' \in \mathbb{Z}_N, \vec{\eta} \in \mathbb{Z}_N^{n+1}$ and R, $R_i \in \mathbb{G}_{p_3}$ for each attribute $i \in S$. It creates the following secret key

$$\overrightarrow{K_1} = < (g_1^{\hat{r}} B_2)^{z_1'}, \cdots, (g_1^{\hat{r}} B_2)^{z_n'}, g_1^{\alpha} (U_1 g_3^{\hat{s}})^t \prod_{i=1}^{n} (g_1^{\hat{r}} B_2)^{-x_i z_i'} > * g_3^{\vec{\eta}}, L = g_1^t R, K_i = T_i^t R_i].$$

Challenge: To form the challenge ciphertext, $\mathcal{B}$ chooses random values $v_2', \cdots, v_{n_2}', r_x' \in \mathbb{Z}_N$. It then creates the vector $\vec{v}' = (1, v_2', \cdots, v_{n_2}')$ and sets

$$C \xleftarrow{R} \mathbb{G}_T, \overline{C_1} = < (g_1^s B_{24})^{x_1}, \cdots, (g_1^s B_{24})^{x_n}, g_1^s B_{24} > * g_4^{\vec{d}}, \; C_x = T^{a A_x^* \vec{v}'} T_{\rho^*(x)}^{-r_x} W_x, D_x = g_1^{r_x} V_x.$$

Phase 2: $\mathcal{B}$ works in the same way as Phase 1.

If $T \xleftarrow{R} U_1^s D_{24}$, the ciphertext components of C is random and $\mathcal{B}$ simulates $Game_{final0}$. Otherwise if $T \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_4}$, the ciphertext components of C, C_x are random and $\mathcal{B}$ simulates $Game_{final1}$. Thus $Game_{final0}$ and $Game_{final1}$ are indistinguishable.

Corollary 1 (Corollary 6.3 in [21])**:** Let $m \in \mathbb{N}$, $m \geq 3$, and let p be a prime. Let $\vec{\delta} \xleftarrow{R} \mathbb{Z}_p^m$, $\vec{\tau} \xleftarrow{R} \mathbb{Z}_p^m$, and let $\vec{\tau}'$ be chosen uniformly randomly from the set of vectors in $\mathbb{Z}_p^m$ which are orthogonal to $\vec{\delta}$ under the dot product modulo p. Let $f : \mathbb{Z}_p^m \to W$ be some function. Then $dist((\vec{\delta}, f(\vec{\tau}')), (\vec{\delta}, f(\vec{\tau}))) \leq \epsilon$, as long as $|W| \leq 4(1 - \frac{1}{p}) p^{m-2} \epsilon^2$, where $dist(Y_1, Y_2)$ denote the statistical distance of two random variables Y_1, Y_2 and $|W|$ is the number of elements of W.

The kth created key is normal or nominal semi-functional for $\vec{\gamma}\vec{\delta} - u_1 \theta \equiv 0 \bmod p_2$. If we set $\vec{\tau}$ to the semi-functional parameters and $\vec{\tau} = \vec{\gamma}, m = n + 1, \epsilon = p_2^{-c}$, then from Corollary 1, the leakage bound of our scheme $l = (n - 2c - 1)\log p_2$ where c is any fixed positive constant.

Theorem 1: Under Assumptions 1, 2, 3, 4 and for $l = (n - 1 - 2c)\log(p_2)$, where $c > 0$ is a fixed positive constant, the proposed scheme is anonymous and l – leakage-resilient.

***Proof*:** If Assumptions 1, 2, 3 and 4 hold, by the previous lemmas we know that the real security game is indistinguishable from $Game_{final1}$, so β is information-theoretically hidden from the adversary. Hence the adversary cannot gain a non-negligible advantage in breaking the anonymous CP-ABE scheme.

5. PERFORMATION ANALYSIS

In this section, the performance comparison in theoretically analysis and practical computation is presented. Table 1 gives the properties comparison with other schemes. The efficiency results of the new systems compared to the old ones are shown in Table 2.

From Table 1, one can find that the schemes [15-17] achieve anonymity in the leak-free scenario and the scheme [17] is fully secure. In addition, the scheme [21] achieves adaptive security in the context of key leakage attacks, but it cannot protect the privacy of users. However our scheme can avoid the weakness above and obtain anonymity and leakage-resilience in the standard model. In Table 2, R_T and R denote the number of bits

for the representation of elements of $\mathbb{G}_T$ and $\mathbb{G}$ respectively. $|S|$ is the number of attributes of keys. The scheme [15] used AND gate as access policy and there are m attributes in universe. Table 2 shows that proposed scheme preserves the efficiency of the original scheme [21]. Compared to the scheme [15], our scheme achieves leakage-resilience, and simultaneously do not sacrifice the efficiency.

Table 1. Properties comparison of different schemes.

Schemes	Access policy	Anonymity	Security	Leakage-resilient
[15]	AND gate	Yes	Selective	No
[16]	AND gate	Yes	Selective	No
[17]	LSSS	Yes	Full	No
[21]	LSSS	No	Full	Yes
Ours	LSSS	Yes	Full	Yes

Table 2. Efficiency comparison with other schemes.

Schemes	CT size	SK size	Anonymity	Leakage-resilient
[15]	$2R_T+(4+3m)R$	$(2+5m)R$	Yes	No
[21]	$R_T+(n+1+2n_1)R$	$(n+2+\|S\|)R$	No	Yes
Ours	$R_T+(n+1+2n_1)R$	$(n+2+\|S\|)R$	Yes	Yes

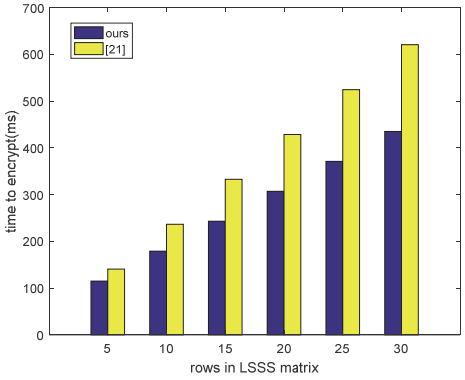

Fig. 1. The comparison of encryption time.

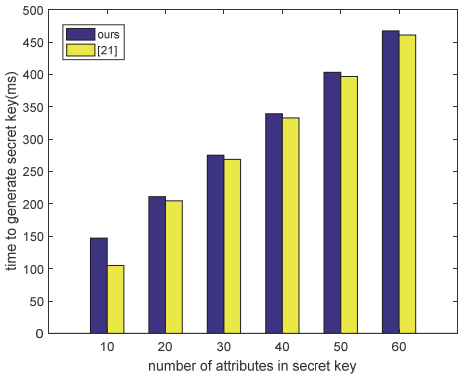

Fig. 2. The comparison of key generation time.

Furthermore, we now provide some information on the performance achieved by PBC library [24]. For encryption and key generation time, only the dominant operations are counted, which are the exponentiations in $\mathbb{G}$ and $\mathbb{G}_T$. Figs. 1 and 2 display measurements of encryption time and secret key generation time, respectively. The experiment is simulated on a modern workstation with 64-bit, 3.2 Ghz Pentium 4. The implementation uses a 160-bit elliptic curve group over a 512-bit finite field. Fig. 1 shows that the encryption time in the proposed construction is less than the scheme [21], and Fig. 2 shows that the key generation time is close to its. It is obvious that only the proposed construction enjoys the desirable property of leakage-resilience and anonymity, and simultaneously don not sacrifice the efficiency.

6. CONCLUSIONS

The anonymous ABE can be applied to hide the receivers' attribute information in ciphertexts. But most of the existing anonymous ABE schemes are considered in the leak-free scenario assuming that secret keys are not leaked to the adversary. In this paper, we proposed a novel anonymous CP-ABE scheme against side-channel attacks which achieves leakage-resilience in the bounded-leakage model. In particular, experimental results show that the proposed solution is efficient and practical. Under some static assumptions, the scheme is fully secure. A drawback of our construction is that it uses bilinear groups of composite order. An open problem is to build such a scheme in symmetric bilinear groups of prime order.

REFERENCES

1. A. Sahai and B. S. Waters, "Fuzzy identity-based encryption," in *Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques*, 2005, pp. 457-473.
2. V. Goyal, O. Pandey, and A. Sahai, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, 2006, pp. 89-98.
3. A. Kapadia, P. Tsang, and S. W. Smith, "Attribute-based publishing with hidden credentials and hidden policies," in *Proceedings of Network and Distributed System Security Symposium*, Vol. 7, 2007, pp. 179-192.
4. J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Proceedings of the 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2008, pp. 146-162.
5. T. Nishide, K. Yoneyama, and K. Ohta, "Abe with partially hidden encryptor-specified access structure," in *Proceedings of the 6th International Conference on 6th International Conference*, 2008, pp. 111-129.
6. J. Li, K. Ren, B. Zhu, and Z. Wan, "Privacy-aware attribute-based encryption with user accountability," in *Proceedings of the 12th International Conference on Information Security*, 2009, pp. 347-362.
7. R. Canetti, Y. Dodis, S. Halevi, E. Kushilevitz, and A. Sahai, "Exposure-resilient functions and all-or-nothing transforms," in *Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques*, 2000, pp. 453-469.
8. S. Faust, T. Rabin, L. Reyzin, E. Tromer, and V. Vaikuntanathan, "Protecting circuits from leakage: the computationally-bounded and noisy cases," in *Proceedings of EUROCRYPT*, 2010, pp. 135-156.
9. Y. Dodis and K. Pietrzak, "Leakage-resilient pseudorandom functions andside-channel attacks on feistel networks," in *Proceedings of Annual Cryptology Conference*, 2010, pp. 21-40.
10. Y. Dodis, S. Goldwasser, Y. Kalai, C. Peikert, and V. Vaikuntanathan, "Public-key encryption schemes with auxiliary inputs," in *Proceedings of the 7th Theory of Cry-*

ptography Conference, 2010, pp. 361-381.
11. T. Yuen, S. Chow, Y. Zhang, and S. Yiu, "Identity-based encryption resilient to continual auxiliary leakage," in *Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2012, pp. 117-134.
12. Z. Wang and S. Yiu, "Attribute-based encryption resilient to auxiliary input," in *Proceedings of the 9th International Conference on Provable Security*, 2015, pp. 371-390.
13. S. Sun, D. Gu, and S. Liu, "Efficient leakage-resilient identity-based encryption with CCA security," in *Proceedings of the 6th International Conference on Pairing-Based Cryptography*, 2013, pp. 149-167.
14. J. Li, M. Teng, Y. Zhang, and Q. Yu, "A leakage-resilient CCA-secure identity-based encryption scheme," *The Computer Journal*, Vol. 59, 2016, pp. 1066-1075.
15. Y. Zhang, X. Chen, J. Li, D. S. Wong, and H. Li, "Anonymous attribute-based encryption supporting efficient decryption test," in *Proceedings of the 8th ACM SIGSAC Symposium on Information*, *Computer and Communications Security*, 2008, pp. 511-516.
16. P. Chaudhari, M. Das, and A. Mathuria, "On anonymous attribute based encryption," in *Proceedings of the 11th International Conference on Information Systems Security*, 2015, pp. 378-392.
17. F. Han, J. Qin, H. Zhao, *et al*., "A general transformation from KP-ABE to searchable encryption," *Future Generation Computer Systems*, Vol. 30, 2014, pp. 107-115.
18. Y. Chen, Z. Zhang, D. Lin, *et al.*, "Anonymous identity-based hash proof system and its applications," in *Proceedings of the 6th International Conference on Provable Security*, 2012, pp. 143-160.
19. L. Zhang, Z. Wang, and Q. Wu, "Leakage-resilient anonymous identity-based broadcast encryption in the standard model," in *Proceedings of International Workshops and Symposiums on Algorithms and Architectures for Parallel Processing*, 2015, pp. 201-210.
20. C. Hu, R. Yang, P. Liu, *et al*., "Public-key encryption with keyword search secure against continual memory attacks," *Security and Communication Networks*, Vol. 9, 2016, pp. 1613-1629.
21. A. Lewko, Y. Rouselakis, and B. Waters, "Achieving leakage resilience through dual system encryption," in *Proceedings of the 8th Theory of Cryptography Conference*, 2011, pp. 70-88.
22. A. Beimel, "Secure schemes for secret sharing and key distribution," Ph.D. Thesis, Faculty of Computer Science, Technion-Israel Institute of Technology.
23. D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Proceedings of the 2nd Theory of Cryptography Conference*, 2005, pp. 325-341.
24. F. Li and W. Wu, *Pairing-Based Cryptography*, Science Press, Beijing, 2014.

Jing-Xia Zhang (张靜霞**)** is a master candidate in the School of Mathematics and Statistics, Xidian University. Her research interests include leakage-resilient cryptography, attribute-based encryption.

Le-You Zhang (张乐友**)** received his Ph.D. from the Xidian University in 2009. Now he is a Professor in the School of Mathematics and Statistics, Xidian University of P.R. China. His current research interests include network security, computer security, and cryptography. He has published more than 70 papers in journals and conferences.