Attribute-Based Data Sharing with Flexible and Direct Revocation in Cloud Computing

Yinghui Zhang[1, 2], Xiaofeng Chen[3], Jin Li[4], Hui Li[3] and Fenghua Li[2]
[1] National Engineering Laboratory for Wireless Security,
Xi'an University of Posts and Telecommunications, Xi'an 710121, P.R. China
[e-mail: yhzhaang@163.com]
[2] State Key Laboratory of Information Security, Institute of Information Engineering,
Chinese Academy of Sciences, Beijing 100093, P.R. China
[e-mail: lfh@iie.ac.cn]
[3] State Key Laboratory of Integrated Service Networks (ISN),
Xidian University, Xi'an 710071, P.R. China
[e-mail: xfchen@xidian.edu.cn; lihui@mail.xidian.edu.cn]
[4] School of Computer Science, Guangzhou University, Guangzhou 510006, P.R. China
[e-mail: jinli71@gmail.com]
*Corresponding author: Yinghui Zhang

Received May 28, 2014; revised August 3, 2014; revised September 13, 2014; accepted October 1, 2014; published November 30, 2014

Abstract

Attribute-based encryption (ABE) is a promising cryptographic primitive for implementing fine-grained data sharing in cloud computing. However, before ABE can be widely deployed in practical cloud storage systems, a challenging issue with regard to attributes and user revocation has to be addressed. To our knowledge, most of the existing ABE schemes fail to support flexible and direct revocation owing to the burdensome update of attribute secret keys and all the ciphertexts. Aiming at tackling the challenge above, we formalize the notion of ciphertext-policy ABE supporting flexible and direct revocation (FDR-CP-ABE), and present a concrete construction. The proposed scheme supports direct attribute and user revocation. To achieve this goal, we introduce an auxiliary function to determine the ciphertexts involved in revocation events, and then only update these involved ciphertexts by adopting the technique of broadcast encryption. Furthermore, our construction is proven secure in the standard model. Theoretical analysis and experimental results indicate that FDR-CP-ABE outperforms the previous revocation-related methods.

Keywords: Data sharing, attribute-based encryption, revocation, cloud computing

A preliminary version of this paper appeared in IEEE INCoS 2013, September 9-11, Xi'an, China. This version includes the description of system architecture, an analysis of design goals, a detailed security proof, experimental results based on PBC library, and extensive performance comparisons. This research was supported by the National Natural Science Foundation of China [61402366, 61272037, 61272455, 61272457, 61472091, 61472472], the Ministry of Industry and Information Key Project Fund [2013ZX03002004], the National Science and Technology Major Projects [2012ZX03002003], Doctoral Fund of Ministry of Education of China [20130203110004], the Fundamental Research Funds for the Central Universities [BDY151402], and the Natural Science Foundation of Shaanxi Province [2013JZ020].

http://dx.doi.org/10.3837/tiis.2014.11.021

1. Introduction

With the advent of cloud computing technology, sharing data through a third-party service provider has never been more economical and convenient than now. However, due to data outsourcing and untrusted storage servers, data access control becomes a challenging issue in cloud storage, where differentiated data access is frequently required in the sense that users with different attributes should be granted different levels of access privileges. Traditional methods based on access control lists are no longer suitable for cloud computing, because they require a fully trusted cloud server.

Aiming at providing fine-grained access control over cloud storage, a novel public key primitive namely attribute-based encryption (ABE) [1] was introduced in the cryptographic community, which enables public one-to-many encryption. ABE comes in two flavors called key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE) [2]. Compared with KP-ABE, CP-ABE is extremely suitable for cloud-based data sharing, because it enables data owners to make and enforce access policies themselves. In CP-ABE, every ciphertext is associated with an access policy, and every secret key is associated with a set of attributes. A particular attribute secret key can decrypt a ciphertext if and only if the attributes associated with the secret key match the underlying access policy in the ciphertext.

Though CP-ABE is a promising primitive for designing fine-grained access control systems in cloud computing, there are several challenges that remain in applications of CP-ABE.

- On the one hand, revocation issues are essential and difficult in CP-ABE systems in that users may change their attributes frequently in practice and each attribute is conceivably shared by multiple users. Most of the existing CP-ABE schemes [3][4][5][6][7][8] only provide indirect revocation mechanisms, which suffer a severe efficiency drawback due to a key update phase. Direct revocation does not require users to update attribute secret keys periodically. However, existing CP-ABE schemes [9][10] only support direct user revocation, and hence cannot realize flexible attribute revocation. In particular, direct attribute revocation can realize a fine-grained revocation mechanism without affecting any non-involved users and hence is a preferable solution. Therefore, one challenge is how to realize direct attribute revocation in CP-ABE.
- On the other hand, CP-ABE has a drawback that ciphertext length often grows with the complexity of access policies [3][10][11][12]. The drawback appears more serious for application scenarios where bandwidth issues are major concerns. Therefore, another challenge is how to keep ciphertext length of CP-ABE constant.

To the authors' knowledge, however, there are no CP-ABE schemes, which have constant-size ciphertexts and provide direct attribute revocation mechanisms.

1.1 Our Contribution

Research contributions of this paper can be summarized as follows:

- Firstly, we analyze security and efficiency goals of attribute-based data sharing in cloud computing. We formalize the notion of CP-ABE supporting flexible and direct revocation mechanisms (FDR-CP-ABE) in the setting of cloud computing, formulate a reasonable security model, and present a concrete construction. The proposed scheme is a directly revocable CP-ABE scheme, which supports direct user and attribute revocation and is applicable to data sharing architectures in cloud computing.

- Secondly, in order to realize a flexible and direct revocation mechanism in the proposed FDR-CP-ABE scheme, we introduce an auxiliary function to specify which ciphertexts are involved in revocation events, and then use the technique of broadcast encryption to only update these involved ciphertexts. In addition, it is shown that our technique is also applicable to KP-ABE counterparts.
- Thirdly, the proposed FDR-CP-ABE scheme is proven secure in the standard model. It achieves the security goals of data confidentiality, collusion-resistance, backward secrecy, and forward secrecy. Theoretical analysis and experimental results indicate that the proposed FDR-CP-ABE outperforms the previous revocation-related methods. In particular, it enjoys desirable properties such as no secret key update, partial ciphertext update, and constant-size ciphertexts.

1.2 Organization

The remaining of this work is organized as follows. In Section 2, we review the state-of-the-art attribute-based encryption schemes. Some preliminaries are given in Section 3. We formalize the notion and security model of FDR-CP-ABE in Section 4. Our FDR-CP-ABE construction is detailed in Section 5. Security results together with performance comparisons are presented in Section 6. In Section 7, the application of our technique to KP-ABE counterparts is discussed. Finally, we conclude this paper in Section 8.

2. Related Work

Since the introduction of ABE [1] in implementing fine-grained data access control systems, plenty of researches have been done on ABE. In KP-ABE, access policies are enforced in secret keys and ciphertexts are labeled with a set of attributes. In CP-ABE, the roles of the attribute set and the access policy are swapped from what we described for KP-ABE. The first KP-ABE construction [2] realized monotonic access structures for key policies. To enable more flexible access policies, Ostrovsky et al. [13] presented the first KP-ABE system that supports the expression of non-monotone formulas in key policies. On the other hand, Bethencourt et al. [3] proposed the first CP-ABE scheme, but the security proof is given in the generic group model. To overcome this weakness, Cheung and Newport [11] presented another construction that is proven selectively secure in the standard model. To achieve full security, Lewko et al. [12] proposed a fully secure CP-ABE scheme in composite order bilinear groups, and proved its security from three static assumptions. There are also many works proposed to make further improvements on ABE, such as accountable ABE [14][15], anonymous ABE [16][17][18], ABE with constant-size ciphertexts [19][20][21], etc. Despite various attractive features, the above CP-ABE schemes cannot realize a revocation mechanism, which is indispensable for attribute-based systems in that users' secret keys might get compromised at some point in the future.

In order to deal with the challenging revocation issue in attribute-based systems, several attribute-revocable ABE schemes have been proposed [3][4]. These schemes realize attribute revocation by setting an expiration time on each attribute, and hence the method is called a **timed rekeying mechanism**. However, these attribute-revocable ABE schemes suffer a security drawback in terms of the backward and forward secrecy, and the method based on validation time fails to realize attribute change in a timely fashion, i.e., the immediate attribute revocation. For the sake of practical ABE systems [5][6][7][8], Yu et al. [5] proposed a CP-ABE scheme supporting immediate attribute revocation mechanisms with the help of a

semi-trusted proxy server. Hur et al. [6] proposed an immediate attribute revocation mechanism in CP-ABE by allowing a proxy server to re-encrypt ciphertexts with a set of attribute group keys. Yang et al. [7] proposed an attribute revocation method to cope with the dynamic changes of users' access privileges. Li et al. [8] used ABE to realize secure sharing of personal health records and their solution supports attribute revocation. Researches on the security of e-healthcare have also been done in [22][23]. However, all the above schemes only support **indirect revocation**, that is, the attribute center indirectly realizes revocation by only allowing non-revoked users to update secret keys. The indirect revocation method has a disadvantage that the key update phase can be a performance bottleneck for both the attribute center and all the non-revoked users.

To tackle the above issue, Attrapadung et al. [9] proposed directly user-revocable CP-ABE schemes by combining the techniques of ABE and broadcast encryption (BE). **Direct revocation** has a desirable property that revocation can be realized without affecting any non-involved users, that is, it does not require users to update attribute secret keys periodically. Since Fiat et al. [24] first introduced the notion of BE, Boneh et al. [25] proposed a collusion-resistant BE scheme with short ciphertexts and private keys. The methods in [9] require that data owners should take full charge of maintaining the membership lists for each attribute group. Accordingly, these schemes are not suitable for data sharing in cloud computing, where the data owners upload their data into clouds and they will no longer be in direct control of the data. Sahai et al. [10] presented a generic method to show that a CP-ABE scheme with ciphertext delegation and piecewise key generation implies a revocable storage CP-ABE scheme. Furthermore, they proposed a variant of the CP-ABE scheme [12] that supports ciphertext delegation and piecewise key generation. However, the proposed scheme fails to support direct attribute revocation and the ciphertext length is not constant. Other researches on direct revocation mechanisms can be seen in [26][27]. The above directly revocable ABE schemes cannot efficiently realize attribute and user revocation, and the ciphertext size linearly increases with the number of revoked users or the complexity of access policies. In the extended abstract [28] of this paper, we formalized the notion of FDR-CP-ABE and presented a concrete scheme. We revise the paper a lot and add more technical details as compared to [28]. Firstly, in order to realize data sharing based on ABE in cloud computing, we add Section 4.2 to describe the system architecture, and add Section 4.3 to analyze security and efficiency goals of attribute-based data sharing systems. Secondly, for the FDR-CP-ABE construction, we provide detailed security proofs in the standard model in Section 6.1. Thirdly, we do intensive experiments and present more extensive performance comparisons in Section 6.2. Lastly, we add Section 7 to demonstrate that our technique is applicable to the KP-ABE counterpart.

3. Preliminaries

3.1 Bilinear Pairings

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic multiplicative groups of prime order p, g be a generator of $\mathbb{G}$, and 1 be the identity of $\mathbb{G}_T$. We call map $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ a bilinear pairing if it satisfies the following properties. 1) **Computability:** there exists an efficient algorithm for computing map $\hat{e}$. 2) **Bilinearity:** $\hat{e}(g^a, g^b) = \hat{e}(g, g)^{ab}$ for all $a, b \in \mathbb{Z}_p^*$. 3) **Non-degeneracy:** $\hat{e}(g, g) \neq 1$.

3.2 Complexity Assumptions

Bilinear Diffie-Hellman Exponent (BDHE) assumption: Let $\mathbb{G}$ be a bilinear group of prime order p, and g, h be two independent generators of $\mathbb{G}$. Let $\overrightarrow{y}_{g,\alpha,\ell} = (g_1, g_2, \cdots, g_\ell, g_{\ell+2}, \cdots, g_{2\ell}) \in \mathbb{G}^{2\ell-1}$, where $g_i = g^{(\alpha^i)}$ for some unknown $\alpha \in \mathbb{Z}_p^*$. An algorithm $\mathcal{B}$ that outputs $\mu \in \{0,1\}$ has advantage ϵ in solving the decision ℓ-BDHE problem if

$$|\Pr[\mathcal{B}(g, h, \overrightarrow{y}_{g,\alpha,\ell}, \hat{e}(g_{\ell+1}, h)) = 1] - \Pr[\mathcal{B}(g, h, \overrightarrow{y}_{g,\alpha,\ell}, Z) = 1]| \geq \epsilon.$$

We say the decision (t, ϵ, ℓ)-BDHE assumption holds in $\mathbb{G}$ if no t-time algorithm has advantage at least ϵ in solving the decision ℓ-BDHE problem in $\mathbb{G}$.

4. Definition and Models

4.1 Notations

For simplicity, we explain some notations in **Table 1**, which are frequently used in this paper. Note that the attribute center in a data sharing system will publish an attribute revocation list on a public bulletin board when an attribute revocation event occurs. In **Table 1**, the attribute revocation information $\mathcal{R}$ and the public parameter PP are published on the public bulletin board by the attribute center.

Table 1. Notations frequently used in this paper

Notations	Description
$x \in_R X$	the element x is chosen from the set X at random.
N_{now}	the number of attribute revocation lists published by the attribute center to date.
$\mathcal{R}_\omega^{(i)}$	the index set of users whose attribute ω has been revoked when the i-th attribute revocation event occurs.
$\mathcal{R}_{\omega^+}^{(i)}/\mathcal{R}_{\omega^-}^{(i)}$	users in $\mathcal{R}_\omega^{(i)}$ whose attribute ω has the occurrence ω^+/ω^- and $\mathcal{R}_\omega^{(i)} = \{\mathcal{R}_{\omega^+}^{(i)}, \mathcal{R}_{\omega^-}^{(i)}\}$.
$\mathcal{R}^{(i)}$	it indicates that the i-th attribute revocation event occurs and $\mathcal{R}^{(i)} = \cup_\omega \mathcal{R}_\omega^{(i)}$.
$\mathcal{R}_W^{(i)}$	the index set of users associated with the access policy W involved in $\mathcal{R}^{(i)}$.
$\mathcal{R}$	N_{now} attribute revocation lists and $\mathcal{R} = \{\mathcal{R}^{(i)}\}_{1 \leq i \leq N_{now}}$.
$PP^{(i)}$	the public parameter corresponding to the i-the attribute revocation event.
PP	N_{now} public parameters corresponding to $\mathcal{R}$ and $PP = \{PP^{(i)}\}_{1 \leq i \leq N_{now}}$.
$S \models W$	the attribute set S satisfies the access policy W.
$S \not\models W$	the attribute set S does not match the access policy W.
SK_S	an attribute secret key associated with the attribute set S.
CT_W	a ciphertext with respect to the access policy W.
$UK^{(i)}$	the ciphertext update key corresponding to $\mathcal{R}^{(i)}$.
CT'_W	an updated ciphertext of CT_W with the access policy W unchanged.

4.2 System Architecture

As shown in **Fig. 1**, the architecture of an attribute-based data sharing system in cloud computing consists of four types of parties: an attribute center, a cloud service provider, data owners, and users. Data owners and users are administrated by the attribute center. The cloud

service provider is honest-but-curious and it manages a cloud to provide data storage service. Note that the cloud is assumed to have sufficient storage capacity and computation power. Data owners encrypt their contents and store ciphertext data in the cloud for sharing. To access the shared contents in the cloud, users download encrypted contents of interest from the cloud and then decrypt them based on their secret keys. In particular, the cloud service provider can update ciphertexts involved in some revocation events based on the delegation key from the attribute center.

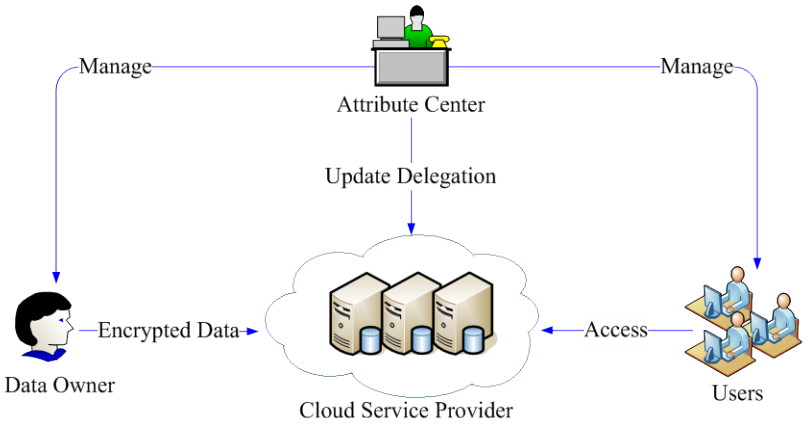


Fig. 1. Architecture of an attribute-based data sharing system

4.3 Design Goals

We aim to propose a CP-ABE scheme supporting flexible and direct revocation mechanisms. On the one hand, it achieves the following security goals.

- **Data Confidentiality.** Unauthorized users who do not have enough attributes matching the access policy specified for a ciphertext by a data owner should be prevented from accessing the plaintext of this ciphertext. In particular, unauthorized access from the cloud service provider to the plaintext should also be prevented.
- **Collusion-Resistance.** If multiple users collude, they may be able to access the plaintext of a given ciphertext by combining their attributes even if each of them cannot decrypt the ciphertext alone. For access control systems in practice, these colluders should not succeed.
- **Backward and Forward Secrecy.** Backward secrecy means that a newly joined user who has sufficient attributes should be able to decrypt the ciphertexts which were published before he holds the attributes. And, forward secrecy means that any user who is involved in a revocation event cannot access the plaintexts of the subsequent ciphertexts exchanged after he drops related attributes, unless the other attributes still satisfy the access policy.

On the other hand, the proposed scheme enjoys the following efficiency benefits.

- **No Secret Key Update.** All the users need not to update attribute secret keys whenever a revocation event occurs. Notice that direct revocation mechanisms enjoy this property.
- **Partial Ciphertext Update.** When an attribute revocation event occurs, the cloud service provider only needs to update partial ciphertexts of which the underlying access policies are involved in the revocation event[1].

[1] Given an access structure W and the attribute revocation information to date denoted by $\mathcal{R}$, we check whether W is involved in $\mathcal{R}$ or not based on an auxiliary function RevoIndex, which is introduced in Section 5.2. Similarly, we

- **Constant-Size Ciphertexts.** The length of a ciphertext is constant and it does not linearly increase with the number of attributes in universe or the number of revocation events.

4.4 Definition of FDR-CP-ABE

A FDR-CP-ABE scheme consists of six algorithms: **Setup**, **KeyGen**, **Encrypt**, **UKeyGen**, **CTUpdate**, and **Decrypt**, where **Encrypt** and **CTUpdate** play an important role in realizing revocation mechanisms. Particularly, there are four types of ciphertexts in FDR-CP-ABE: Type-1 ciphertexts, Type-2 ciphertexts, Type-3 ciphertexts, and Type-4 ciphertexts, which are defined in the following algorithms **Encrypt** and **CTUpdate**. It is worth noting that Type-1 and Type-2 ciphertexts are generated by encryptors in the algorithm **Encrypt**, while Type-3 and Type-4 ciphertexts are generated by cloud service providers in the algorithm **CTUpdate**.

▶ $\mathbf{Setup}(1^\lambda) \rightarrow (PK, MK)$: On input a security parameter λ, it returns the system public key PK which is distributed to users, and the master key MK which is kept private by the attribute center.

▶ $\mathbf{KeyGen}(PK, MK, S) \rightarrow SK_S$: On input PK, MK and an attribute set S, it outputs the attribute secret key SK_S associated with the set S.

▶ $\mathbf{Encrypt}(PK, M, W, \mathcal{R}) \rightarrow CT_W$: On input PK, a message M, an access structure W, and the attribute revocation information $\mathcal{R}$ to date, it generates a ciphertext CT_W of M with respect to W.

Remark 1. We say CT_W is a Type-1 ciphertext if W is not involved in $\mathcal{R}$. Otherwise, CT_W is said to be a Type-2 ciphertext if W is involved in $\mathcal{R}$. Simply speaking, Type-1 ciphertexts are not involved in revocation events while Type-2 ciphertexts are relevant to revocation events. In the concrete scheme in Section 5.3, if a user is involved in any one of revocation events in $\mathcal{R}$, he fails to recover M from $CT_W = \mathsf{Encrypt}(PK, M, W, \mathcal{R})$ even if his attribute set satisfies W. Hence, Encrypt plays a role of attribute revocation.

▶ $\mathbf{UKeyGen}\big(PK, MK, \mathcal{R}^{(k)}\big) \rightarrow \big(PP^{(k)}, UK^{(k)}\big)$: On input PK, MK, and an attribute revocation list $\mathcal{R}^{(k)}$ published by the attribute center when the k-th revocation event occurs, it generates the public parameter $PP^{(k)}$ and ciphertext update key $UK^{(k)}$ corresponding to $\mathcal{R}^{(k)}$. The attribute center publishes $PP^{(k)}$ on a public bulletin board, and sends $UK^{(k)}$ to the cloud service provider through a secure channel.

▶ $\mathbf{CTUpdate}\big(PK, CT_W, UK^{(k)}, \mathcal{R}^{(k)}\big) \rightarrow CT'_W$: On input PK, CT_W with respect to W, $UK^{(k)}$ and $\mathcal{R}^{(k)}$, it generates an updated ciphertext CT'_W of CT_W with W unchanged if and only if W is involved in $\mathcal{R}^{(k)}$. It needs not to update CT_W if W is not involved in $\mathcal{R}^{(k)}$.

Remark 2. We say CT'_W is a Type-3 (resp. Type-4) ciphertext if CT_W is a Type-1 (resp. Type-2) ciphertext. Furthermore, if CT_W is a Type-3 (resp. Type-4) ciphertext, the updated ciphertext CT'_W is still a Type-3 (resp. Type-4) ciphertext. Simply speaking, Type-3 ciphertexts are geneated by updating Type-1 or Type-3 ciphertexts, and Type-4 ciphertexts are geneated by updating Type-2 or Type-4 ciphertexts. In the concrete scheme in Section 5.3, if a user is involved in $\mathcal{R}^{(k)}$, he fails to decrypt $CT'_W = \mathsf{CTUpdate}\big(PK, CT_W, UK^{(k)}, \mathcal{R}^{(k)}\big)$ even if he can decrypt CT_W. Hence, CTUpdate plays a role of attribute revocation.

---

can check whether W is involved in the i-th attribute revocation list $\mathcal{R}^{(i)}$ or not based on RevoIndex.

▶ **Decrypt**$(PK, PP, CT_W, SK_S) \rightarrow M$ or $\perp$: On input PK, the public parameters PP corresponding to all the attribute revocation events to date, a ciphertext CT_W of a message M under the access policy W, and a secret key SK_S associated with the attribute set S, it checks if $S \models W$ and SK_S is not involved in attribute revocation events associated with CT_W. If so, it returns message M. Otherwise, it returns $\perp$ with overwhelming probability.

4.5 Security Model

In order to achieve the security goals considered in Section 4.3, we model the capability of adversaries. We formalize two types of adversaries: Type-I adversary $\mathcal{A}_I$ and Type-II adversary $\mathcal{A}_{II}$. $\mathcal{A}_I$ aims to break the confidentiality of Type-1 ciphertexts in which no attribute revocation events are involved, and hence $\mathcal{A}_I$ is not allowed to make a secret key query on the attribute set satisfying the challenge access structure. However, $\mathcal{A}_{II}$ intends to break the confidentiality of Type-2, Type-3, and Type-4 ciphertexts, which are involved in revocation events, and hence $\mathcal{A}_{II}$ is allowed to make a secret key query on any attribute sets. It is worth observing that the design goals of *Data Confidentiality*, *Collusion-Resistance* and *Backward and Forward Secrecy* are integrated in the indistinguishability against selective ciphertext-policy and chosen plaintext attacks (IND-sCP-CPA) model, which is based on the following IND-sCP-CPA game involving an adversary $\mathcal{A}_i(i = I, II)$ and a simulator $\mathcal{B}$. In fact, in the initialization phase of the proposed security model, $\mathcal{A}_I$ only needs to submit a challenge access structure W^* to the simulator, and $\mathcal{A}_{II}$ has to additionally submit attribute revocation information $\mathcal{R}^*$ and an attribute revocation list $\mathcal{R}^{*(k)}$. In order to integrate collusion-resistance, different users are allowed to collude to guess the random bit chosen by the challenger in the security model. To demonstrate that backward and forward secrecy is reflected in the security model, different kinds of challenge ciphertexts are generated based on $\mathcal{R}^*$ and $\mathcal{R}^{*(k)}$ in the challenge phase. Hence, if the proposed scheme is proven secure in the proposed security model, it enjoys data confidentiality, collusion-resistance and backward and forward secrecy. The IND-sCP-CPA game is described as follows:

▶ **Init:** $\mathcal{A}_i(i = I, II)$ chooses a challenge access structure W^* and submits it to $\mathcal{B}$. It should be noted that attribute revocation information is published on a public bulletin board by $\mathcal{B}$. In addition, $\mathcal{A}_{II}$ submits attribute revocation information $\mathcal{R}^* = \{\mathcal{R}^{*(1)}, \mathcal{R}^{*(2)}, \cdots, \mathcal{R}^{*(j)}\}$ and an attribute revocation list $\mathcal{R}^{*(k)}$ with $k \geq j + 1$.

▶ **Setup:** $\mathcal{B}$ chooses a security parameter λ, and runs the Setup algorithm to get a master key SK and the corresponding system public key PK. It retains SK and gives PK to $\mathcal{A}_i$.

▶ **Phase 1:** $\mathcal{A}_i$ issues a polynomially (in λ) bounded number of queries as follows:

- KeyGen oracle $\mathcal{O}_{KeyGen}$: $\mathcal{A}_i$ submits an attribute set S, and $\mathcal{B}$ answers queries from $\mathcal{A}_I$ and $\mathcal{A}_{II}$ as follows:
  - ➢ For $\mathcal{A}_I$, $\mathcal{B}$ returns SK_S with a restriction that $S \not\models W^*$, and returns $\perp$ if $S \models W^*$.
  - ➢ For $\mathcal{A}_{II}$, $\mathcal{B}$ returns SK_S even if $S \models W^*$.
- UKeyGen oracle $\mathcal{O}_{UKeyGen}$: $\mathcal{A}_i$ submits an attribute revocation list $\mathcal{R}^{(k)}$, and $\mathcal{B}$ returns the ciphertext update key $UK^{(k)}$ corresponding to $\mathcal{R}^{(k)}$.
- CTUpdate oracle $\mathcal{O}_{CTUpdate}$: $\mathcal{A}_i$ submits a ciphertext CT_W, and attribute revocation list $\mathcal{R}^{(k)}$ published by the attribute center. $\mathcal{B}$ returns an updated ciphertext CT'_W of CT_W.

▶ **Challenge:** Once $\mathcal{A}_i$ decides that Phase 1 is over, it outputs two equal length messages

M_0, M_1, on which it wishes to be challenged with respect to W^*. $\mathcal{B}$ chooses a bit $b \in_R \{0,1\}$, and generates challenge ciphertexts for $\mathcal{A}_i$ as follows:

- For $\mathcal{A}_I$, $\mathcal{B}$ returns $CT_{W^*} = \mathsf{Encrypt}(PK, M_b, W^*, \emptyset)$, where CT_{W^*} is of Type-1.
- For $\mathcal{A}_{II}$, we consider three circumstances.
  - **Case 1.** W^* is involved in $\mathcal{R}^*$. In this case, $\mathcal{B}$ returns $CT_{W^*} = \mathsf{Encrypt}(PK, M_b, W^*, \mathcal{R}^*)$, and hence the challenge ciphertext CT_{W^*} is of Type-2.
  - **Case 2.** W^* is not involved in $\mathcal{R}^*$, but it is involved in $\mathcal{R}^{*(k)}$. In this case, $\mathcal{B}$ returns
    $$CT'_{W^*} = \mathsf{CTUpdate}\left(PK, CT_{W^*}, UK^{(k)}, \mathcal{R}^{*(k)}\right),$$
    where $CT_{W^*} = \mathsf{Encrypt}(PK, M_b, W^*, \mathcal{R}^*)$, and hence CT'_{W^*} is of Type-3.
  - **Case 3.** W^* is involved in $\mathcal{R}^*$ and $\mathcal{R}^{*(k)}$ simultaneously. In this case, $\mathcal{B}$ computes CT'_{W^*} as in Case 2, and returns CT'_{W^*} as the challenge ciphertext, which is of Type-4.

▶ **Phase 2:** The same as Phase 1. Furthermore, $\mathcal{A}_i$ can make ciphertext update queries on challenge ciphertexts.

▶ **Guess:** $\mathcal{A}_i$ outputs a guess bit $b' \in \{0,1\}$ and wins the game if $b' = b$. The advantage of $\mathcal{A}_i$ in the above IND-sCP-CPA game is defined as $\mathsf{Adv}^{\text{IND-sCP-CPA}}_{\text{FDR-CP-ABE}}(\mathcal{A}_i) = \left|\Pr[b' = b] - \frac{1}{2}\right|$.

Definition 1. *A probabilistic algorithm $\mathcal{A}$ is said to (t, ϵ, q_K)-break a FDR-CP-ABE scheme if $\mathcal{A}$ achieves an advantage $\mathsf{Adv}^{\text{IND-sCP-CPA}}_{\text{FDR-CP-ABE}}(\mathcal{A}) \geq \epsilon$, when running in at most t steps, and making at most q_K queries to the key generation oracle $\mathcal{O}_{KeyGen}$. A FDR-CP-ABE scheme is said to be (t, ϵ, q_K)-secure if no forger can (t, ϵ, q_K)-break it.*

5. Construction of FDR-CP-ABE

5.1 Attribute and Access Structure

Suppose there are n attributes in universe denoted by $\mathcal{U} = \{\omega_1, \omega_2, \cdots, \omega_n\}$ for a certain natural number n. And, each attribute ω_i would have three occurrences: positive ω_i^+, negative ω_i^- and "don't care" $*$, where ω_i^+ represents a user has the attribute ω_i, and ω_i^- denotes a user does not have ω_i or ω_i is not a proper attribute of the user. We consider the access structure W that consists of AND gates on positive and negative attributes, that is, $W = \wedge_{i \in \mathcal{I}_W} \overline{\omega}_i$, where $\mathcal{I}_W \subseteq \{1, 2, \cdots, n\}$ is the index set of attributes specified in W and $\overline{w}_i$ is ω_i^+ or ω_i^-. If an attribute does not appear in the AND gate, its occurrence is "don't care". This kind of policies are also adopted in [5][11]. It is noted that $S \models W$ if and only if for $i \in \mathcal{I}_W$, $\omega_i \in S$ when $\overline{\omega}_i = \omega_i^+$ and $\omega_i \notin S$ when $\overline{\omega}_i = \omega_i^-$.

5.2 Auxiliary Function

We introduce an auxiliary function RevoIndex to check whether an access structure W is involved in an attribute revocation list $\mathcal{R}^{(k)}$ or not. In other words, we can decide based on RevoIndex if a ciphertext with the underlying access structure W should be updated when the k-th attribute revocation event occurs.

$\mathbf{RevoIndex}(PK, W, \mathcal{R}^{(k)}) \rightarrow \mathcal{R}_W^{(k)}$: On input PK, W and $\mathcal{R}^{(k)}$, RevoIndex outputs the index set $\mathcal{R}_W^{(k)}$ associated with W of users involved in the k-th attribute revocation event.

Note that $\mathcal{R}^{(k)} = \{\mathcal{R}_{\omega}^{(k)} | \omega \in \mathcal{U}^{(k)}\}$, where $\mathcal{U}^{(k)} = \{\omega_i | i \in \mathcal{I}_{\mathcal{U}^{(k)}}\} \subseteq \mathcal{U}$ and $\mathcal{R}_{\omega}^{(k)} = \{\mathcal{R}_{\omega^+}^{(k)}, \mathcal{R}_{\omega^-}^{(k)}\}$. $\mathcal{U}^{(k)}$ is the set of attributes the attribute center has revoked. Let $\mathcal{I}_{\mathcal{R}_W^{(k)}} = \mathcal{I}_W \cap \mathcal{I}_{\mathcal{U}^{(k)}}$, then RevoIndex outputs $\mathcal{R}_W^{(k)} = \cup_{i \in \mathcal{I}_{\mathcal{R}_W^{(k)}}} \mathcal{R}_{\overline{\omega}_i}^{(k)}$, where $\mathcal{R}_{\overline{\omega}_i}^{(k)} = \mathcal{R}_{\omega_i^+}^{(k)}$ if $\overline{\omega}_i = \omega_i^+$ and $\mathcal{R}_{\overline{\omega}_i}^{(k)} = \mathcal{R}_{\omega_i^-}^{(k)}$ if $\overline{\omega}_i = \omega_i^-$. Suppose $\mathcal{R}_W^{(k)}$=RevoIndex$(PK, W, \mathcal{R}^{(k)})$. If $\mathcal{R}_W^{(k)} = \emptyset$, the ciphertexts under W have not to be updated even if the k-th attribute revocation event occurs. Otherwise, $\mathcal{R}_W^{(k)} \neq \emptyset$, the ciphertexts under W have to be updated by the attribute center such that users specified by $\mathcal{R}_W^{(k)}$ cannot access these ciphertexts again.

For a better understanding, we illustrate RevoIndex by an example. As shown in **Table 2**, we consider $n = 10$, $m = 20$, $W = \omega_1^+ \wedge \omega_2^- \wedge \omega_4^+$ and $\mathcal{R}^{(k)} = \{\mathcal{R}_{\omega_1}^{(k)}, \mathcal{R}_{\omega_2}^{(k)}, \mathcal{R}_{\omega_3}^{(k)}\}$, it easily follows that $\mathcal{I}_W = \{1, 2, 4\}$, $\mathcal{U}^{(k)} = \{\omega_1, \omega_2, \omega_3\}$ and $\mathcal{I}_{\mathcal{U}^{(k)}} = \{1, 2, 3\}$. Hence, we have $\mathcal{I}_{\mathcal{R}_W^{(k)}} = \mathcal{I}_W \cap \mathcal{I}_{\mathcal{U}^{(k)}} = \{1, 2\}$ and $\mathcal{R}_W^{(k)} = \mathcal{R}_{\overline{\omega}_1}^{(k)} \cup \mathcal{R}_{\overline{\omega}_2}^{(k)} = \mathcal{R}_{\omega_1^+}^{(k)} \cup \mathcal{R}_{\omega_2^-}^{(k)}$. From **Table 2**, where RSN deontes the revocation serial number, we know that $\mathcal{R}_W^{(k)} = \{1, 2, 5, 8\}$. That is, when the k-th attribute revocation event occurs, the ciphertexts under W have to be updated such that users specified by $\mathcal{R}_W^{(k)} = \{1, 2, 5, 8\}$ cannot access them again.

Table 2. Data structure of the attribute revocation list $\mathcal{R}^{(k)}$

RSN	Involved attributes	Occurrence	Involved users
k	ω_1	ω_1^+	$\mathcal{R}_{\omega_1^+}^{(k)} = \{1, 8\}$
		ω_1^-	$\mathcal{R}_{\omega_1^-}^{(k)} = \{2, 3, 5, 7\}$
	ω_2	ω_2^+	$\mathcal{R}_{\omega_2^+}^{(k)} = \{1, 4, 6, 9\}$
		ω_2^-	$\mathcal{R}_{\omega_2^-}^{(k)} = \{2, 5, 8\}$
	ω_3	ω_3^+	$\mathcal{R}_{\omega_3^+}^{(k)} = \emptyset$
		ω_3^-	$\mathcal{R}_{\omega_3^-}^{(k)} = \{9, 10\}$

5.3 Construction

▶ **Setup**(1^λ)**:** Let $\mathbb{G}, \mathbb{G}_T$ be cyclic multiplicative groups of prime order p, and $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be a bilinear map. Define a hash function $H : \{1, 2, \cdots, 2n\} \to \mathbb{Z}_p^*$. The attribute center chooses a generator $g \in_R \mathbb{G}$, $x_1, x_2, \cdots, x_{2n} \in_R \mathbb{Z}_p^*$ and $y_1, y_2, \cdots, y_{2n} \in_R \mathbb{Z}_p^*$. For $i = 1, 2, \cdots, 2n$, the attribute center sets $u_i = g^{-x_i}$, $Y_i = \hat{e}(g, g)^{y_i H(i)}$. It also picks $\alpha, \beta \in_R \mathbb{Z}_p^*$ and sets $v = g^\beta$. Suppose the total number of users in the system is bounded above by some natural number m. For notational simplicity, we let $\mathcal{I}_m = \{1, 2, \cdots, m\}$ in the following. For $i = 1, 2, \cdots, m, m+2, m+3, \cdots, 2m$, the attribute center computes $g_i = g^{(\alpha^i)}$. The system public key is published as $PK = \langle g, \{u_k, Y_k\}_{1 \le k \le 2n}, \{g_k\}_{1 \le k \le 2m, k \neq m+1}, v \rangle$. The master key is $MK = \langle \{x_k, y_k\}_{1 \le k \le 2n}, \beta \rangle$.

▶ **KeyGen**(PK, MK, S)**:** Let S be an attribute set of the user who wants to obtain the corresponding attribute secret key. The attribute center chooses $h \in_R \mathbb{G}$ for the user. Then for $i \in \{1, 2, \cdots, n\}$, it computes $\overline{\sigma}_i$ as follows:

$$\overline{\sigma}_i = \begin{cases} \sigma_i = g^{y_i H(i)} h^{x_i}, & \text{if } w_i^+ \in S, \\ \sigma_{i+n} = g^{y_{i+n} H(i+n)} h^{x_{i+n}}, & \text{if } w_i^- \in S. \end{cases}$$

Also, the attribute center computes $d = g_{sn}^{\beta}$, where $sn \in \{1, 2, \cdots, m\}$ is a serial number. Note that sn is used by the attribute center to indicate that the current user is the sn-th one to join the system. Finally, the attribute secret key is $SK_S = \langle sn, h, \{\overline{\sigma}_i\}_{1 \le i \le n}, d \rangle$.

► $\mathbf{Encrypt}(PK, M, W, \mathcal{R})$**:**[2] Suppose the attribute center has published a total of N_{now} attribute revocation lists denoted by $\mathcal{R}$. We have $\mathcal{R} = \{\mathcal{R}^{(i)}\}_{1 \le i \le N_{now}}$, where $\mathcal{R}^{(i)}$ represents the i-th attribute revocation list. In order to encrypt a message $M \in \mathbb{G}_T$ under a ciphertext policy $W = \wedge_{i \in \mathcal{I}_W} \overline{w}_i$, an encryptor computes $\langle u_W, Y_W \rangle = \langle \prod_{i \in \mathcal{I}_W} \overline{u}_i, \prod_{i \in \mathcal{I}_W} \overline{Y}_i \rangle$, where $\langle \overline{u}_i, \overline{Y}_i \rangle$ is defined as follows:

- If $\overline{\omega}_i = \omega_i^+$, then $\langle \overline{u}_i, \overline{Y}_i \rangle = \langle u_i, Y_i \rangle = \langle g^{-x_i}, \hat{e}(g,g)^{y_i H(i)} \rangle$.
- If $\overline{\omega}_i = \omega_i^-$, then $\langle \overline{u}_i, \overline{Y}_i \rangle = \langle u_{i+n}, Y_{i+n} \rangle = \langle g^{-x_{i+n}}, \hat{e}(g,g)^{y_{i+n} H(i+n)} \rangle$.

In addition, for $1 \le i \le N_{now}$, the encryptor uses W and $\mathcal{R}^{(i)}$ to call RevoIndex to generate $\mathcal{R}_W^{(i)} = \mathsf{RevoIndex}(PK, W, \mathcal{R}^{(i)})$. Then, it sets $\mathcal{R}_W = \cup_{i=1}^{N_{now}} \mathcal{R}_W^{(i)}$, where $\mathcal{R}_W$ represents the attribute revocation information corresponding to W in $\mathcal{R}$. The encryptor chooses $s \in_R \mathbb{Z}_p^*$ and computes the ciphertext CT_W of M with respect to W as follows:

- If $\mathcal{R}_W = \emptyset$, then a Type-1 ciphertext is generated. In this case, no revocation information of W exists currently, and the encryptor sets the ciphertext as $CT_W = \langle W, C_0, C_1, C_2 \rangle$, where $C_0 = MY_W^s$, $C_1 = g^s$, and $C_2 = u_W^s$.
- If $\mathcal{R}_W \neq \emptyset$, then a Type-2 ciphertext is generated. In this case, some revocation information to date is related to W, and the encryptor computes $K_{\mathcal{R}} = \hat{e}(g_1, g_m)^s$, $C_{\mathcal{R}} = \left(v \cdot \prod_{i \in \mathcal{I}_m - \mathcal{R}_W} g_{m+1-i} \right)^s$. Then it sets $C_0 = MY_W^s K_{\mathcal{R}}$, $C_1 = g^s$, $C_2 = u_W^s$. Finally, $CT_W = \langle W, C_0, C_1, C_2, C_{\mathcal{R}} \rangle$.

► $\mathbf{UKeyGen}\left(PK, MK, \mathcal{R}^{(k)}\right)$**:** The attribute center chooses $uk^{(k)} \in_R \mathbb{Z}_p^*$, sets $UK^{(k)} = uk^{(k)}\beta$ and computes $PP^{(k)} = v^{uk^{(k)}} = g^{UK^{(k)}}$. Then, it publishes $PP^{(k)}$ on a public bulletin board, and sends $UK^{(k)}$ to the cloud service provider through a secure channel.

► $\mathbf{CTUpdate}\left(PK, CT_W, UK^{(k)}, \mathcal{R}^{(k)}\right)$**:** In order to update the ciphertext CT_W according to the k-th attribute revocation list $\mathcal{R}^{(k)}$, in the following, four circumstances are taken into consideration in terms of the type of CT_W.

- **Case 1.** $CT_W = \langle W, C_0, C_1, C_2 \rangle$ is a Type-1 ciphertext generated by encryptors. In this case, we know $k = 1$. For $1 \le i \le k$, the cloud service provider compute
$$\mathcal{R}_W^{(i)} = \mathsf{RevoIndex}(PK, W, \mathcal{R}^{(i)}).$$
Subsequently, it sets $\mathcal{R}_W^{(k)} = \mathcal{R}_W^{(k)} - \bigcup_{i=1}^{k-1} \mathcal{R}_W^{(i)}$, where $\mathcal{R}_W^{(0)} = \emptyset$. Then, if $\mathcal{R}_W^{(k)} = \emptyset$, there is no need to update. Otherwise, $\mathcal{R}_W^{(k)} \neq \emptyset$, the cloud service provider computes

[2] We denote by $\mathcal{R}_u$ the index set of revoked users at some point. To realize user revocation on the system level, the encryptor just set $\mathcal{R}_W = \mathcal{R}_W \cup \mathcal{R}_u$ in the algorithm Encrypt. In addition, the cloud service provider can perform the algorithm CTUpdate based on each $\mathcal{R}^{(i)}$ determined by $\mathcal{R}_u$. That is, in any case, the revoked users are eliminated from the broadcast set.

$K = \hat{e}(g_1, g_m)^{UK^{(k)}}$. Then it sets $C_0' = C_0 \cdot K$, and computes $C_{\widehat{\mathcal{R}}} = C_{\mathcal{R}^{(k)}}$, where

$$C_{\mathcal{R}^{(k)}} = \left(v \cdot \prod\nolimits_{i \in \mathcal{I}_m - \mathcal{R}_W^{(k)}} g_{m+1-i} \right)^{UK^{(k)}}.$$

Finally, $CT_W' = \langle W, C_0', C_1, C_2, C_{\widehat{\mathcal{R}}} \rangle$, which is said to be a Type-3 ciphertext.

- **Case 2.** $CT_W = \langle W, C_0, C_1, C_2, C_{\mathcal{R}} \rangle$ is a Type-2 ciphertext generated by encryptors. Suppose $\mathcal{R} = \{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \cdots, \mathcal{R}^{(j)}\}$, we know $j \geq 1$ and $k = j + 1$. In this case, the cloud service provider generates the ciphertext components C_0' and $C_{\widehat{\mathcal{R}}}$ as in Case 1. Finally, $CT_W' = \langle W, C_0', C_1, C_2, C_{\mathcal{R}}, C_{\widehat{\mathcal{R}}} \rangle$, which is said to be a Type-4 ciphertext.
- **Case 3.** $CT_W = \langle W, C_0, C_1, C_2, C_{\widehat{\mathcal{R}}} \rangle$ is a Type-3 ciphertext generated by the cloud service provider. In this case, $k \geq 2$. For $1 \leq i \leq k$, the cloud service provider computes $\mathcal{R}_W^{(i)} = \mathsf{RevoIndex}(PK, W, \mathcal{R}^{(i)})$. Subsequently, it sets $\mathcal{R}_W^{(k)} = \mathcal{R}_W^{(k)} - \bigcup_{i=1}^{k-1} \mathcal{R}_W^{(i)}$. Then, if $\mathcal{R}_W^{(k)} = \emptyset$, there is no need to update. Otherwise, $\mathcal{R}_W^{(k)} \neq \emptyset$, the cloud service provider computes $K = \hat{e}(g_1, g_m)^{UK^{(k)}}$. It sets $C_0' = C_0 \cdot K$ and $C_{\widehat{\mathcal{R}}}' = C_{\widehat{\mathcal{R}}} \cdot C_{\mathcal{R}^{(k)}}$, where

$$C_{\mathcal{R}^{(k)}} = \left(v \cdot \prod\nolimits_{i \in \mathcal{I}_m - \mathcal{R}_W^{(k)}} g_{m+1-i} \right)^{UK^{(k)}}.$$

Finally, $CT_W' = \langle W, C_0', C_1, C_2, C_{\widehat{\mathcal{R}}}' \rangle$, which is still a Type-3 ciphertext.

- **Case 4.** $CT_W = \langle W, C_0, C_1, C_2, C_{\mathcal{R}}, C_{\widehat{\mathcal{R}}} \rangle$ is a Type-4 ciphertext from the cloud service provider. Suppose $\mathcal{R} = \{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \cdots, \mathcal{R}^{(j)}\}$, we know $j \geq 1$ and $k \geq j + 2$. In this case, the cloud service provider updates ciphertext components C_0 and $C_{\widehat{\mathcal{R}}}$ as in Case 3. Finally, $CT_W' = \langle W, C_0', C_1, C_2, C_{\mathcal{R}}, C_{\widehat{\mathcal{R}}}' \rangle$, which is still a Type-4 ciphertext.

► **Decrypt**(PK, PP, CT_W, SK_S)**:** The ciphertext CT_W can be decrypted by a user with secret key $SK_S = \langle sn, h, \{\overline{\sigma}_i\}_{1 \leq i \leq n}, d \rangle$ as follows. If $S \not\models W$, the algorithm returns $\perp$. Otherwise, $S \models W$, there are four cases in terms of the type of CT_W to be considered.

- **Case 1.** For a Type-1 ciphertext $CT_W = \langle W, C_0, C_1, C_2 \rangle$, compute $\sigma_W = \prod_{i \in \mathcal{I}_W} \overline{\sigma}_i$, and the message is recovered as

$$M = \frac{C_0}{\hat{e}(\sigma_W, C_1)\hat{e}(h, C_2)}.$$

- **Case 2.** For a Type-2 ciphertext $CT_W = \langle W, C_0, C_1, C_2, C_{\mathcal{R}} \rangle$, suppose $\mathcal{R} = \{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \cdots, \mathcal{R}^{(j)}\}$, we know $j \geq 1$. Then, for $1 \leq i \leq j$, the user computes $\mathcal{R}_W^{(i)} = \mathsf{RevoIndex}(PK, W, \mathcal{R}^{(i)})$. Subsequently, it sets $\mathcal{R}_W = \bigcup_{i=1}^{j} \mathcal{R}_W^{(i)}$. If $sn \in \mathcal{R}_W$, return $\perp$. Otherwise, the user computes $\sigma_W = \prod_{i \in \mathcal{I}_W} \overline{\sigma}_i$ and

$$K_{\mathcal{R}} = \frac{\hat{e}\left(g_{sn}, C_{\mathcal{R}}\right)}{\hat{e}\left(d \cdot \prod_{i \in \mathcal{I}_m - \mathcal{R}_W}^{i \neq sn} g_{m+1-i+sn}, C_1\right)}.$$

Finally, the message can be recovered as

$$M = \frac{C_0}{\hat{e}(\sigma_W, C_1)\hat{e}(h, C_2)K_{\mathcal{R}}}.$$

- **Case 3.** The ciphertext $CT_W = \langle W, C_0, C_1, C_2, C_{\widehat{\mathcal{R}}} \rangle$ is of Type-3. Suppose $\mathcal{R}^{(N_{now})}$ is the latest revocation list published by the attribute center. For $1 \leq i \leq N_{now}$, the user

computes $\mathcal{R}_W^{(i)} = \mathsf{RevoIndex}(PK, W, \mathcal{R}^{(i)})$. It sets $\mathcal{R}_W = \bigcup_{i=1}^{N_{now}} \mathcal{R}_W^{(i)}$. If $sn \in \mathcal{R}_W$, the algorithm returns $\perp$. Otherwise, the user computes $\sigma_W = \prod_{i \in \mathcal{I}_W} \overline{\sigma}_i$ and

$$K_{\widehat{\mathcal{R}}} = \frac{\hat{e}\left(g_{sn}, C_{\widehat{\mathcal{R}}}\right)}{\prod_{k=1}^{N_{now}} \hat{e}\left(d \cdot \prod_{i \in \mathcal{I}_m - \mathcal{R}_W^{(k)}}^{i \neq sn} g_{m+1-i+sn}, PP^{(k)}\right)}.$$

Finally, the message can be recovered as

$$M = \frac{C_0}{\hat{e}(\sigma_W, C_1)\hat{e}(h, C_2)K_{\widehat{\mathcal{R}}}}.$$

- **Case 4.** For a Type-4 ciphertext $CT_W = \langle W, C_0, C_1, C_2, C_{\mathcal{R}}, C_{\widehat{\mathcal{R}}} \rangle$, suppose $\mathcal{R}^{(N_{now})}$ is the latest revocation list published by the attribute center and $\mathcal{R} = \{\mathcal{R}^{(1)}, \mathcal{R}^{(2)}, \cdots, \mathcal{R}^{(j)}\}$, we know that $j \geq 1$and $j+1 \leq N_{now}$. For $1 \leq i \leq N_{now}$, the user computes $\mathcal{R}_W^{(i)} = \mathsf{RevoIndex}(PK, W, \mathcal{R}^{(i)})$. Then it sets $\mathcal{R}_W = \bigcup_{i=1}^{j} \mathcal{R}_W^{(i)}$ and $\widehat{\mathcal{R}}_W = \bigcup_{i=1}^{N_{now}} \mathcal{R}_W^{(i)}$. If $sn \in \widehat{\mathcal{R}}_W$, the algorithm returns $\perp$. Otherwise, the user computes $\sigma_W = \prod_{i \in \mathcal{I}_W} \overline{\sigma}_i$,

$$K_{\mathcal{R}} = \frac{\hat{e}\left(g_{sn}, C_{\mathcal{R}}\right)}{\hat{e}\left(d \cdot \prod_{i \in \mathcal{I}_m - \mathcal{R}_W}^{i \neq sn} g_{m+1-i+sn}, C_1\right)},$$

$$K_{\widehat{\mathcal{R}}} = \frac{\hat{e}\left(g_{sn}, C_{\widehat{\mathcal{R}}}\right)}{\prod_{k=j+1}^{N_{now}} \hat{e}\left(d \cdot \prod_{i \in \mathcal{I}_m - \mathcal{R}_W^{(k)}}^{i \neq sn} g_{m+1-i+sn}, PP^{(k)}\right)}.$$

Finally, the message can be recovered as

$$M = \frac{C_0}{\hat{e}(\sigma_W, C_1)\hat{e}(h, C_2)K_{\mathcal{R}}K_{\widehat{\mathcal{R}}}}.$$

6. Analysis of the Proposed FDR-CP-ABE Scheme

6.1 Security Analysis

Theorem 1. Suppose the decision (t, ϵ, m)-BDHE assumption holds in $\mathbb{G}$, then the proposed FDR-CP-ABE scheme is (t, ϵ, m)-secure, where m is an upper bound of the total number of users in the system.

Proof. Suppose there exists a t-time adversary $\mathcal{A}$ $(\mathcal{A}_I, \mathcal{A}_{II})$ such that $\mathsf{Adv}_{\mathsf{FDR\text{-}CP\text{-}ABE}}^{\mathsf{IND\text{-}sCP\text{-}CPA}}(\mathcal{A}) \geq \epsilon$. We build a simulator $\mathcal{B}$ that has advantage ϵ in solving the decision m-BDHE problem in $\mathbb{G}$. $\mathcal{B}$ takes as input a random decision m-BDHE challenge $(g, \overline{h}, \overrightarrow{y}_{g,\alpha,m}, Z))$, where $\overrightarrow{y}_{g,\alpha,m} = (g_1, g_2, \cdots, g_m, g_{m+2}, \cdots, g_{2m})$ and Z is either $\hat{e}(g_{m+1}, \overline{h})$ or a random element in $\mathbb{G}_T$. The simulator $\mathcal{B}$ plays a role of the challenger in the IND-sCP-CPA game, and interacts with the adversary $\mathcal{A}$ $(\mathcal{A}_I, \mathcal{A}_{II})$ as follows.

▶ **Init.** The simulator $\mathcal{B}$ receives a challenge access structure $W^* = \wedge_{i \in \mathcal{I}_{W^*}} \overline{\omega}_i$ specified by the adversary $\mathcal{A}$ $(\mathcal{A}_I, \mathcal{A}_{II})$, where $\mathcal{I}_{W^*} = \{i_1, i_2, \cdots, i_w\} \subseteq \mathbb{Z}_p^*$ with $\omega \leq n$ represents the attribute index set specified in the challenge access structure W^*. In addition, $\mathcal{A}_{II}$ submits attribute revocation information $\mathcal{R}^* = \{\mathcal{R}^{*(1)}, \mathcal{R}^{*(2)}, \cdots, \mathcal{R}^{*(j)}\}$ and an attribute revocation list $\mathcal{R}^{*(k)}$ with $k \geq j+1$.

▶ **Setup.** $\mathcal{B}$ chooses $j^* \in_R \{1, 2, \cdots, w\}$, $x_{i_j} \in_R \mathbb{Z}_p^*$ for $i_j \in \mathcal{I}_{W^*}$, and $x'_k, y_k \in_R \mathbb{Z}_p^*$ for

$1 \le k \le 2n$. In the following, to generate components $\{u_\ell | 1 \le \ell \le 2n\}$, there are three cases to be considered.

- For $i_j \in \mathcal{I}_{W^*} - \{i_{j^*}\}$, $\mathcal{B}$ does the following:
  - If $\overline{w}_{i_j} = w_{i_j}^+$, computes
$$\begin{cases} \left(u_{i_j}, Y_{i_j}\right) = \left(g^{-x_{i_j}} g_{m+1-i_j}^{-1}, \hat{e}(g,g)^{y_{i_j} H(i_j)}\right), \\ \left(u_{i_j+n}, Y_{i_j+n}\right) = \left(g^{-x'_{i_j+n}}, \hat{e}(g,g)^{y_{i_j+n} H(i_j+n)}\right). \end{cases}$$
  - If $\overline{w}_{i_j} = w_{i_j}^-$, computes
$$\begin{cases} \left(u_{i_j}, Y_{i_j}\right) = \left(g^{-x'_{i_j}}, \hat{e}(g,g)^{y_{i_j+n} H(i_j+n)}\right), \\ \left(u_{i_j+n}, Y_{i_j+n}\right) = \left(g^{-x_{i_j}} g_{m+1-i_j}^{-1}, \hat{e}(g,g)^{y_{i_j} H(i_j)}\right). \end{cases}$$
- For i_{j^*}, $\mathcal{B}$ does the following:
  - If $\overline{w}_{i_{j^*}} = w_{i_{j^*}}^+$, computes
$$\begin{cases} \left(u_{i_{j^*}}, Y_{i_{j^*}}\right) = \left(g^{-x_{i_{j^*}}} \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} g_{m+1-k}, \hat{e}(g,g)^{y_{i_{j^*}} H(i_{j^*})} \hat{e}(g,g)^{\alpha^{m+1}}\right), \\ \left(u_{i_{j^*}+n}, Y_{i_{j^*}+n}\right) = \left(g^{-x'_{i_{j^*}+n}}, \hat{e}(g,g)^{y_{i_{j^*}+n} H(i_{j^*}+n)}\right). \end{cases}$$
  - If $\overline{w}_{i_{j^*}} = w_{i_{j^*}}^-$, computes
$$\begin{cases} \left(u_{i_{j^*}}, Y_{i_{j^*}}\right) = \left(g^{-x'_{i_{j^*}}}, \hat{e}(g,g)^{y_{i_{j^*}+n} H(i_{j^*}+n)}\right), \\ \left(u_{i_{j^*}+n}, Y_{i_{j^*}+n}\right) = \left(g^{-x_{i_{j^*}}} \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} g_{m+1-k}, \hat{e}(g,g)^{y_{i_{j^*}} H(i_{j^*})} \hat{e}(g,g)^{\alpha^{m+1}}\right). \end{cases}$$
- For $\ell \notin \mathcal{I}_{W^*}$, $\mathcal{B}$ computes
$$\begin{cases} (u_\ell, Y_\ell) = \left(g^{-x'_\ell}, \hat{e}(g,g)^{y_\ell H(\ell)}\right), \\ (u_{\ell+n}, Y_{\ell+n}) = \left(g^{-x'_{\ell+n}}, \hat{e}(g,g)^{y_{\ell+n} H(\ell+n)}\right). \end{cases}$$

Furthermore, $\mathcal{B}$ chooses $\beta \in \mathbb{Z}_p^*$, and sets $v = g^\beta \left(\prod_{j \in U^*} g_{m+1-j}\right)^{-1}$ if $\mathcal{R}_{W^*} \neq \emptyset$, where $U^* \subseteq \mathcal{R}_{W^*}$ denotes the target set of involved users to be challenged by the adversary $\mathcal{A}_{II}$ when revocation events occur, else $v = g^\beta$ if $\mathcal{R}_{W^*} = \emptyset$. Then the system public key is $PK = \langle g, \{u_k, Y_k\}_{1 \le k \le 2n}, \{g_k\}_{1 \le k \le 2m, k \neq m+1}, v \rangle$ and $\mathcal{B}$ sends PK to $\mathcal{A}$.

▶ **Phase 1.** The adversary $\mathcal{A}$ ($\mathcal{A}_I$, $\mathcal{A}_{II}$) makes the following queries.

- **KeyGen oracle** $\mathcal{O}_{KeyGen}(S)$: Suppose $\mathcal{A}$ summits an attribute set S in a secret key query. If $S \not\models W^*$, there must exist $i_j \in \mathcal{I}_{W^*}$ such that $\omega_{i_j} \notin \overline{\omega}_{i_j}$. Without loss of generality, we only consider the case of $\omega_{i_j} \notin S$ and $\overline{\omega}_{i_j} = \omega_{i_j}^+$. $\mathcal{B}$ chooses $z \in_R \mathbb{Z}_p^*$ and sets $h = g_{i_j} g^z$. Furthermore, for i_j, $\mathcal{B}$ computes the attribute secret key component as $\overline{\sigma}_{i_j} = g^{y_{i_j+n} H(i_j+n)} (g_{i_j} g^z)^{x'_{i_j+n}}$. For $\ell \neq i_j$, $\mathcal{B}$ computes $\overline{\sigma}_\ell$ in the following:

  Case 1. If $\ell \in \mathcal{I}_{W^*} - \{i_{j^*}\}$, $\mathcal{B}$ computes $\overline{\sigma}_\ell = g^{y_\ell H(\ell)} (g_{i_j})^{x_\ell} g_{m+1-\ell+i_j} (\overline{u}_\ell)^{-z}$.

  Case 2. If $\ell = i_{j^*}$, $\mathcal{B}$ computes $\overline{\sigma}_{i_{j^*}}$ as
$$g^{y_{i_{j^*}} H(i_{j^*})} (g_{i_j})^{x_{i_{j^*}}} \left(\prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}, i_j\}} g_{m+1-k+i_j}^{-1} \right) (\overline{u}_{i_{j^*}})^{-z}.$$

Case 3. If $\ell \notin \mathcal{I}_{W^*}$, $\mathcal{B}$ computes

$$\overline{\sigma}_\ell = \begin{cases} g^{y_\ell H(\ell)}(g_{i_j} g^z)^{x'_\ell} & \text{if } \overline{\omega}_\ell = \omega_\ell^+, \\ g^{y_{\ell+n} H(\ell+n)}(g_{i_j} g^z)^{x'_{\ell+n}} & \text{if } \overline{\omega}_\ell = \omega_\ell^-. \end{cases}$$

Subsequently, if $\mathcal{R}_{W^*} \neq \emptyset$, $\mathcal{B}$ computes $d = g_{sn}^{\beta} \prod_{j \in U^*} g_{m+1-j+sn}^{-1}$. It is noted that

$$d = \left(g^{\beta} \textstyle\prod_{j \in U^*} g_{m+1-j}^{-1}\right)^{(\alpha^{sn})} = v^{(\alpha^{sn})}.$$

If $\mathcal{R}_{W^*} = \emptyset$, $\mathcal{B}$ computes $d = g_{sn}^{\beta} = v^{(\alpha^{sn})}$. The key point is that $sn \leq m$, and that since $sn \in \mathcal{I}_m - \mathcal{R}_{W^*}$ we know $sn \neq j$ and the product defining d does not include the term g_{m+1}. It follows that $\mathcal{B}$ has all the necessary values to compute the secret component d. On the other hand, if $\mathcal{A} = \mathcal{A}_{II}$ and $S \models W^*$, $\mathcal{B}$ chooses $i_j \in_R \mathcal{I}_{W^*}$ and generates a secret key in the method above. In any case, $\mathcal{B}$ returns $SK_S = \langle sn, h, \{\overline{\sigma}_i\}_{1 \leq i \leq n}, d \rangle$.

- **UKeyGen oracle** $\mathcal{O}_{UKeyGen}$: $\mathcal{A}$ submits an attribute revocation list $\mathcal{R}^{(k)}$, and $\mathcal{B}$ chooses $uk^{(k)} \in_R \mathbb{Z}_p^*$, and computes the ciphertext update key $UK^{(k)} = uk^{(k)}\beta$ and $PP^{(k)} = v^{uk^{(k)}} = g^{UK^{(k)}}$ corresponding to $\mathcal{R}^{(k)}$. Then $\mathcal{B}$ returns $UK^{(k)}$ and publishes $PP^{(k)}$ on the public bulletin board.
- **CTUpdate oracle** $\mathcal{O}_{CTUpdate}$: $\mathcal{A}$ submits a ciphertext CT_W, and any attribute revocation list $\mathcal{R}^{(k)}$ published by the attribute center. $\mathcal{B}$ uses $UK^{(k)}$ to generate a updated ciphertext CT'_W of CT_W based on the algorithm definition and returns CT'_W.

▶ **Challenge.** $\mathcal{B}$ runs the IND-sCP-CPA game under the aggregated public encryption key. We denote $x_{W^*} = \sum_{k \in \mathcal{I}_{W^*}} x_k = \sum_{j=1}^{w} x_{i_j}$. Then the aggregated public encryption key is $\langle u_{W^*}, Y_{W^*} \rangle$, where

$$\begin{cases} u_{W^*} = \overline{u}_{i_{j^*}} \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} \overline{u}_k = \left(g^{-x_{i_{j^*}}} \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} g_{m+1-k}\right) \cdot \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} g^{-x_k} g_{m+1-k}^{-1} = g^{-x_{W^*}}, \\ Y_{W^*} = \overline{Y}_{i_{j^*}} \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} \overline{Y}_k = \hat{e}(g,g)^{y_{i_{j^*}} H(i_{j^*})} \hat{e}(g,g)^{\alpha^{m+1}} \cdot \prod_{k \in \mathcal{I}_{W^*} - \{i_{j^*}\}} \hat{e}(g,g)^{y_k H(k)} \\ \qquad = \hat{e}(g,g)^{\sum_{j=1}^{w} y_{i_j} H(i_j) + \alpha^{m+1}}. \end{cases}$$

$\mathcal{B}$ can challenge $\mathcal{A}$ as follows. $\mathcal{A}$ summits two messages M_0 and M_1 of equal length. $\mathcal{B}$ chooses $b \in_R \{0,1\}$, and computes $C_0^* = M_b Y_{W^*}^s = M_b Z^s \hat{e}(g,\overline{h})^{\sum_{j=1}^{w} y_{i_j} H(i_j)}$, $C_1^* = \overline{h}$, and $C_2^* = \overline{h}^{-x_{W^*}}$. Then $\mathcal{B}$ generates challenge ciphertexts for $\mathcal{A}$ as follows:

- For $\mathcal{A}_I$, $\mathcal{B}$ returns $CT_{W^*} = \langle W^*, C_0^*, C_1^*, C_2^* \rangle$. Then, $\mathcal{R}_{W^*} = \emptyset$ and CT_{W^*} is of Type-1.
- For $\mathcal{A}_{II}$, $\mathcal{R}_{W^*} \neq \emptyset$ and there are three circumstances to be considered.

  Case 1. W^* is involved in $\mathcal{R}^*$. In this case, $\mathcal{B}$ computes $K_{\mathcal{R}^*} = Z$, $C_0^* = C_0^* K_{\mathcal{R}^*}$ and $C_{\mathcal{R}^*} = \overline{h}^{\beta} = (g^{\beta})^s = \left(g^{\beta}\left(\prod_{j \in U^*} g_{m+1-j}\right)^{-1}\left(\prod_{j \in U^*} g_{m+1-j}\right)\right)^s = \left(v \cdot \prod_{j \in U^*} g_{m+1-j}\right)^s$. Then it sets $CT_{W^*} = \langle W^*, C_0^*, C_1^*, C_2^*, C_{\mathcal{R}^*} \rangle$, and hence CT'_{W^*} is of Type-2.

  Case 2. W^* is not involved in $\mathcal{R}^*$, but it is involved in $\mathcal{R}^{*(k)}$. In this case, $\mathcal{B}$ returns $CT'_{W^*} = \mathsf{CTUpdate}\left(PK, CT_{W^*}, UK^{(k)}, \mathcal{R}^{*(k)}\right)$, where $CT_{W^*} = \langle W^*, C_0^*, C_1^*, C_2^* \rangle$, and hence CT'_{W^*} is of Type-3.

  Case 3. W^* is not only involved in $\mathcal{R}^*$, but also involved in $\mathcal{R}^{*(k)}$. In this case, $\mathcal{B}$ computes CT'_{W^*} as in Case 2, and it returns CT'_{W^*}, which is of Type-4.

The challenge ciphertext CT'_{W^*} is a valid encryption of M_b whenever $Z = \hat{e}(g_{m+1}, \overline{h})$. On the

other hand, when Z is a random element, CT'_{W^*} is independent of b in the adversary's view.

▶ **Phase 2:** The same as Phase 1. Furthermore, the adversary $\mathcal{A}$ can make ciphertext update queries on challenge ciphertexts.

▶ **Guess:** $\mathcal{A}$ outputs a guess bit b' of b. If $b' = b$, $\mathcal{B}$ outputs 1 in the m-BDHE game to guess that $Z = \hat{e}(g_{m+1}, \overline{h})$. Otherwise, it outputs 0 to indicate that T is a random element in $\mathbb{G}_T$. Note that if $Z = \hat{e}(g_{m+1}, \overline{h})$, then CT_{W^*} is a valid ciphertext and we have

$$\Pr[\,\mathcal{B}(g, \overline{h}, \overrightarrow{y}_{g,\alpha,m}, \hat{e}(g_{m+1}, \overline{h})) = 1\,] = \frac{1}{2} + \mathsf{Adv}^{\mathsf{IND\text{-}sCP\text{-}CPA}}_{\mathsf{FDR\text{-}CP\text{-}ABE}}(\mathcal{A}) \geq \frac{1}{2} + \epsilon.$$

If Z is a random element in $\mathbb{G}_T$, the message M_b is completely hidden from $\mathcal{A}$, and we have

$$\Pr[\,\mathcal{B}(g, \overline{h}, \overrightarrow{y}_{g,\alpha,m}, Z) = 1\,] = \frac{1}{2}.$$

Therefore, it follows that $\mathcal{B}$ has advantage at least ϵ in solving decision m-BDHE in $\mathbb{G}$ within time t. This concludes the proof of Theorem 1.

Remark 3. (A Possible Privacy Leakage) In the proposed security model, the adversaries who are able to learn of some correlations between the previous ciphertext and the updated ciphertext are not taken into consideration. It follows from the proposed scheme that many elements from CT'_W are the same as CT_W, which means some users may learn of the correlation between CT_W and CT'_W. In particular, if revoked users can find this correlation and collude with users who previously decrypt the ciphertexts, they would be able to obtain the plaintexts. So, the proposed scheme seems cannot tackle this kind of privacy leakage. In the proposed construction, it is assumed that the previous ciphertexts are deleted from storage servers by the cloud service provider. The ciphertexts which are involved in revocation events are updated based on the ciphertext update algorithm. Otherwise, revoked users only need to decrypt the previous ciphertexts to obtain corresponding plaintexts in that they have the decryption ability before revocation. On the other hand, in the proposed security model, two kinds of adversaries $\mathcal{A}_I$ and $\mathcal{A}_{II}$ are taken into account. In particular, $\mathcal{A}_{II}$ is allowed to make secret key queries on any attribute sets. In the initialization phase, $\mathcal{A}_{II}$ has to submit attribute revocation information $\mathcal{R}^* = \{\mathcal{R}^{*(1)}, \mathcal{R}^{*(2)}, \cdots, \mathcal{R}^{*(j)}\}$ and $\mathcal{R}^{*(k)}$ with $k \geq j+1$. In the challenge phase, three types of updated ciphertexts are returned to $\mathcal{A}_{II}$ as challenge ciphertexts, which are generated based on the above revocation information. However, $\mathcal{A}_{II}$ fails to guess the random bit chosen by the challenger and hence finds no information about plaintexts from the challenge ciphertexts. In a word, the proposed scheme is proven secure in the proposed security model, and it has some limitations with respect to security considering the above possible privacy leakage.

6.2 Performance Comparison

In this section, we compare the security and efficiency of the proposed FDR-CP-ABE scheme with some existing revocable CP-ABE schemes [3][5][6][9][10]. The notations used in the comparison are described in **Table 3**. In **Table 4**, these schemes are compared with respect to the parameter size, the decryption cost, the type of revocation mechanisms, and the application in the setting of data sharing. It is noted that direct revocation can eliminate the performance bottleneck due to attribute secret key updates. As shown in **Table 4**, only the schemes in [9][10] and ours achieve direct user revocation on the system level, of which only the proposed scheme realizes direct attribute revocation. In particular, the proposed FDR-CP-ABE scheme is a directly revocable CP-ABE scheme applicable to the setting of data sharing.

Table 3. Notations used in comparisons

Notations	Description
L_0	bit size of an element in $\mathbb{G}$.
L_1	bit size of an element in $\mathbb{G}_T$.
L_k	bit size of the attribute set associated with the attribute secret key of a user.
L_{kek}	bit size of a key encryption key (in [3][6]).
t	the number of attributes appeared in an access structure.
ℓ	the number of columns of an access structure (in [10]).
n	the size of the attribute universe.
r	the number of revocation events.
k	the number of attributes associated with the attribute secret key of a user.
m	the maximum number of users in the system.
k_{max}	the maximum size allowed for k (in [9]).
t_{max}	the maximum size allowed for t (in [9]).

Table 4. Security and efficiency comparisons of revocable CP-ABE schemes

Schemes	Parameter size			Decryption cost (pairing)	Direct revocation [3]		Data sharing
	Ciphertext	Attribute Secret Key	System Public Key		User	Attribute	
BSW [3]	$(2t+1)L_0+L_1$	$(2k+1)L_0+L_{kek}$	L_0+L_1	$2k+1$	×	×	✓
YWRL [5]	$(n+1)L_0+L_1$	$(2n+1)L_0+L_k$	$(3n+1)L_0+L_1$	$n+1$	×	×	✓
HN [6]	$(2t+1)L_0+L_1$	$(2k+1)L_0$ $+(\log m)L_{kek}$	L_0+L_1	$2k+1$	×	×	✓
BCP-ABE1 [9]	$(t+2)L_0+L_1$	$(k+2)L_0$	$(k_{max}+t_{max}$ $+2m+1)L_0$	$2t$ $+m+1$	✓	×	×
BCP-ABE1 [9]	$(t+2r+1)L_0$ $+L_1$	$(k+4)L_0$	$(k_{max}+t_{max}$ $+7)L_0+L_1$	$2t$ $+2r+1$	✓	×	×
SSW [10]	$(2\ell+1)L_0+L_1$	$(4t+4)L_0$	$(n+2)L_0+L_1$	$2t+1$	✓	×	✓
Ours FDR-CP-ABE [4]	$\leq 2L_0+2L_1$	$(n+2)L_0+\log m$	$(4n+2m+1)L_0$	$r+2$	✓	✓	✓

On the other hand, ciphertext size implies the communication cost in the system. We note that only the proposed FDR-CP-ABE scheme has constant-size ciphertexts. Furthermore, whenever a revocation event occurs, all the ciphertexts in schemes [3][5][6][9] have to be updated to realize secure access control, while our scheme only needs to update partial ciphertexts which are involved in revocation. Compared with the directly revocable schemes in [9], our FDR-CP-ABE is more efficient in terms of the system public key size and decryption cost. The scheme [10] has two attractive properties: (1) The generality of the proposed method; (2) The support of updating ciphretexts to others with more restrictive access policies. However, the proposed method suffers an efficient drawback in that all the ciphertexts have to be updated whenever a revocation event occurs. In addition, the proposed concrete scheme in [10] fails to support direct attribute revocation and the ciphertext length is not constant. Compared with the scheme [10], our construction is more desirable because it

[3] The schemes [3][5][6] fail to support direct revocation mechanisms.
[4] Only the proposed FDR-CP-ABE scheme enjoys the desirable property of **partial ciphertext update**.

enjoys direct attribute revocation, partial ciphertext update, and constant-size ciphertexts. Our scheme has a disadvantage that it only achieves selective security. In future research, we will focus on directly attribute-revocable CP-ABE schemes with full security. In general, the proposed FDR-CP-ABE scheme is the first CP-ABE scheme supporting flexible and direct attribute revocation, and it has constant-size ciphertexts.

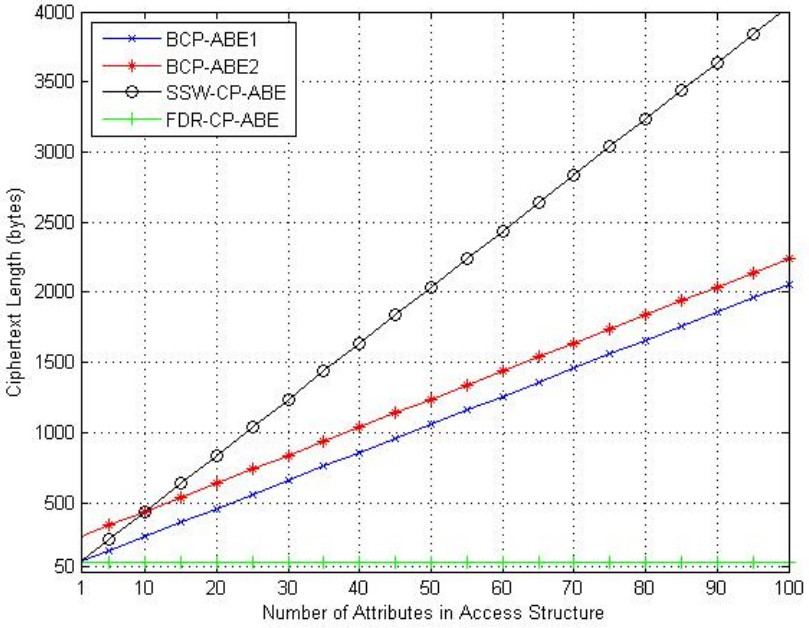


Fig. 2. Comparison of ciphertext length

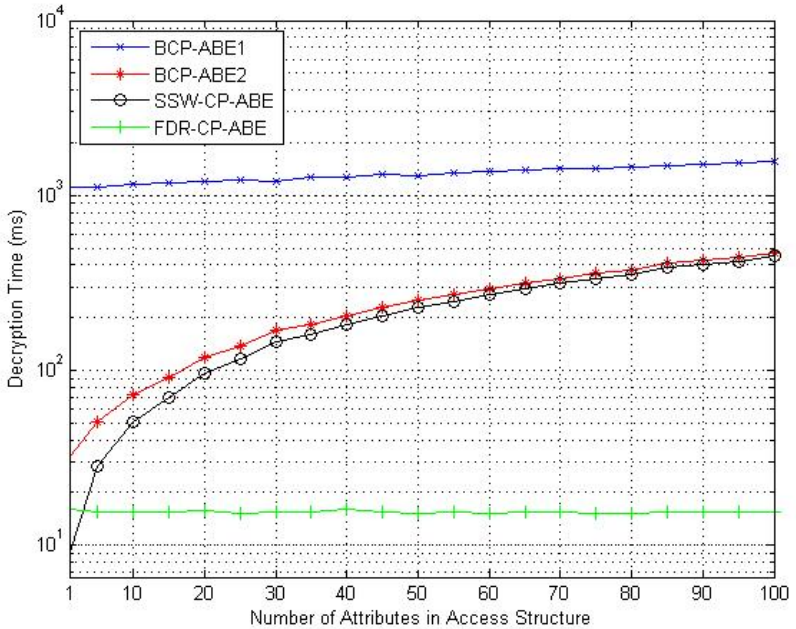


Fig. 3. Comparison of cost for decryption

Considering the desirable properties of direct revocation, we compare schemes [9] denoted as BCP-ABE1 and BCP-ABE2, scheme [10] denoted as SSW-CP-ABE, and ours in terms of the ciphertext length and the decryption cost in **Fig. 2** and **Fig. 3**, respectively. For the ciphertext length comparison, we set $L_0 = L_1 = 160$ bits and the number of revocation events as $r = 5$. Notice that the ciphertext length in the scheme BCP-ABE2 linearly increases with r. In the decryption cost comparison, we set $r = 5$ and the maximum number of users in the system is $m = 500$. In order to precisely evaluate the performance of BCP-ABE1, BCP-ABE2, SSW-CP-ABE, and FDR-CP-ABE, our simulation experiments are based on the Stanford Pairing-Based Crypto library (version 0.5.12) [29] and a Linux machine with 3.30 GHz $\times$ 8 Intel Xeon(R) E3-1230 V2 CPU and 7.5 GB of RAM. In our experiments, we consider the worst case of the access policy, which ensures that all the ciphertext components are involved in decryption. Specifically, we generate 100 distinct access policies in the form of $(\omega_1^+ \wedge \omega_2^+ \wedge \cdots \wedge \omega_t^+)$ with $t(= n)$ increasing from 1 to 100. For each access policy, we repeat the experiment 10 times and take the average values as the final results. Given the number of revocation events, both the decryption cost of the schemes BCP-ABE1, BCP-ABE2, and SSW-CP-ABE is linearly proportional to the number of attributes or columns in access structures, and the decryption cost of ours is constant. Therefore, we argue that the proposed FDR-CP-ABE scheme is more suitable for data sharing in cloud computing.

7. FDR-KP-ABE: KP-ABE with Flexible and Direct Revocation

In this section, we show that the idea of constructing FDR-CP-ABE can be used to realize KP-ABE with flexible and direct revocation (FDR-KP-ABE). In KP-ABE, the roles of the attribute set and access policy are swapped from what we described for CP-ABE. That is, each ciphertext is labeled by the data owner with a set of descriptive attributes, while each secret key is associated with an access policy on attributes that specifies which type of ciphertexts the secret key can decrypt. A particular user can decrypt a particular ciphertext only if the ciphertext attributes satisfy the access policy of the key. An exciting application of KP-ABE is pay-TV systems, in which user access privileges are defined over content attributes and could be determined by the price they paid. In these scenarios, the issue of key revocation also exists. In order to realize flexible and direct revocation, we can introduce an auxiliary function to determine which ciphertext components are involved in some revocation events, and then use the BE technique to update these involved ciphertexts by setting the broadcast set as the index set of non-involved users. In the following, we illustrate the above method by an example.

Suppose a ciphertext corresponds to an attribute set

$$S=\{\text{“CHANNEL: 1”}, \text{“TYPE: SPORT”}, \text{“TYPE: MOVIE”}, \text{“TYPE: NEWS”}\},$$

while a key policy W is associated to TV program package keys that a particular user receives when subscribing programs, where

$$W=\text{“CHANNEL: 1”} \wedge (\text{“TYPE: SPORT”} \vee \text{“TYPE: MOVIE”} \vee \text{“TYPE: MUSIC”} \vee \text{“TYPE: BUSINESS”} \vee \text{“TYPE: NEWS”}).$$

Now, the user is allowed to access any programs of types “SPORT”, “MOVIE”, or “NEWS” provided by channel 1. Later, the system administrator wants to disable the user’s access right on programs with type “SPORT” for some reasons such as unpaid expenses. For this purpose, it is necessary to revoke the corresponding components of the user’s secret key. In fact, the storage server just needs to specify the broadcast set as the index set of all users excluding the revoked one, and then based on the technique of BE to update the ciphertext components associated with the attribute “TYPE: SPORT”.

8. Conclusion

We formalize the notion of FDR-CP-ABE and present a concrete scheme, which is based on AND-gates policy supporting positive and negative attributes with wildcards. The proposed scheme is proven secure and enjoys desirable properties such as no secret key update, partial ciphertext update, and constant-size ciphertexts. The FDR-CP-ABE construction can be used to realize fine-grained attribute-based access control over encrypted data in cloud computing. In addition, we show that our technique is applicable to the KP-ABE counterpart.

References

[1] A. Sahai and B. Waters, "Fuzzy identity-based encryption," *EUROCRYPT'05*, LNCS 3494, pp. 557-557, May 22-26, 2005. Article (CrossRef Link)

[2] V. Goyal, O. Pandey and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. of the 13th ACM conference on Computer and Communications Security (CCS'06)*, pp. 89-98, October 30- November 3, 2006. Article (CrossRef Link)

[3] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. of IEEE Symposium on Security and Privacy (SP'07)*, pp. 321-334, May 20-23, 2007. Article (CrossRef Link)

[4] A. Boldyreva, V. Goyal and V. Kumar, "Identity-based encryption with efficient revocation," in *Proc. of the 15th ACM conference on Computer and communications security (CCS'08)*, pp. 417-426, October 27-31, 2008. Article (CrossRef Link)

[5] S. Yu, C. Wang, K. Ren and W. Lou, "Attribute based data sharing with attribute revocation," in *Proc. of the 5th ACM Symposium on Information Computer and Communications Security (ASIACCS'10)*, pp. 261-270, April 13-16, 2010. Article (CrossRef Link)

[6] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 7, pp. 1214-1221, 2011. Article (CrossRef Link)

[7] K. Yang, X. Jia and K. Ren, "Attribute-based fine-grained access control with efficient revocation in cloud storage systems," in *Proc. of the 8th ACM Symposium on Information, Computer and Communications Security (ASIACCS'13)*, pp. 523-528, May 8-10, 2013. Article (CrossRef Link)

[8] M. Li, S. Yu, Y. Zheng, K. Ren and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131-143, 2013. Article (CrossRef Link)

[9] N. Attrapadung and H. Imai, "Conjunctive broadcast and attribute-based encryption," *Pairing'09*, LNCS 5671, pp. 248-265, August 12-14, 2009. Article (CrossRef Link)

[10] A. Sahai, H. Seyalioglu and B. Waters, "Dynamic credentials and ciphertext delegation for attribute-based encryption," *CRYPTO'12*, LNCS 7417, pp. 199-217, August 19-23, 2012. Article (CrossRef Link)

[11] L. Cheung and C. Newport, "Provably secure ciphertext policy abe," in *Proc. of the 14th ACM conference on Computer and Communications Security (CCS'07)*, pp. 456-465, October 29-November 2, 2007. Article (CrossRef Link)

[12] A. Lewko, T. Okamoto, A. Sahai, K. Takashima and B. Waters, "Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption," *EUROCRYPT'10*, LNCS 6110, pp. 62-91, May 30-June 3, 2010. Article (CrossRef Link)

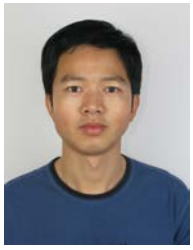
[13] R. Ostrovsky, A. Sahai and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proc. of the 14th ACM conference on Computer and Communications Security (CCS'07)*, pp. 195-203, October 29- November 2, 2007. Article (CrossRef Link)

[14] J. Li, K. Ren, B. Zhu and Z. Wan, "Privacy-aware attribute-based encryption with user accountability," in *Proc. of the International Information Security Conference (ISC'09)*, LNCS 5735, pp. 347-362, September 7-9, 2009. Article (CrossRef Link)

[15] Z. Liu, Z. Cao and D. S. Wong, "Blackbox traceable cp-abe: how to catch people leaking their keys by selling decryption devices on ebay," in *Proc. of the 20th ACM conference on Computer and Communications Security (CCS'13)*, pp. 475-486, November 4-8, 2013. Article (CrossRef Link)
[16] T. Nishide, K. Yoneyama and K. Ohta, "Abe with partially hidden encryptor-specified access structure," in *Proc. of Applied Cryptography and Network Security (ACNS'08)*, LNCS 5037, pp. 111-129, June 3-6, 2008. Article (CrossRef Link)
[17] J. Lai, R. H. Deng and Y. Li, "Expressive cp-abe with partially hidden access structures," in *Proc. of the 7th ACM Symposium on Information, Computer and Communications Security (ASIACCS'12)*, pp. 18-19, May 2-4, 2012. Article (CrossRef Link)
[18] Y. Zhang, X. Chen, J. Li, D. S. Wong and H. Li, "Anonymous attribute-based encryption supporting efficient decryption test," in *Proc. of the 8th ACM Symposium on Information, Computer and Communications Security (ASIACCS'13)*, pp. 511-516, May 8-10, 2013. Article (CrossRef Link)
[19] C. Chen, Z. Zhang and D. Feng, "Efficient ciphertext policy attribute-based encryption with constant-size ciphertext and constant computation-cost," *ProvSec'11*, LNCS 6980, pp. 84-101, October 16-18, 2011. Article (CrossRef Link)
[20] J. Herranz, F. Laguillaumie and C. Ràfols, "Constant size ciphertexts in threshold attribute-based encryption," *PKC'10*, LNCS 6056, pp. 19-34, May 26-28, 2010. Article (CrossRef Link)
[21] A. Ge, R. Zhang, C. Chen, C. Ma and Z. Zhang, "Threshold ciphertext policy attribute-based encryption with constant size ciphertexts," in *ACISP'12*, LNCS 7372, pp. 336-349, July 9-11, 2012. Article (CrossRef Link)
[22] R. Lu, X. Lin and X. Shen, "SPOC: A secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 3, pp. 614-624, 2013. Article (CrossRef Link)
[23] N. D. Han, L. Han, D. M. Tuan, H. P. In and M. Jo, "A scheme for data confidentiality in cloud-assisted wireless body area networks," *Information Sciences*, vol. 284, pp. 157-166, 2014. Article (CrossRef Link)
[24] A. Fiat and M. Naor, "Broadcast encryption," *CRYPTO'93*, LNCS 773, pp. 480-491, August 22-26, 1993. Article (CrossRef Link)
[25] D. Boneh, C. Gentry and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys," *CRYPTO'05*, LNCS 3621, pp. 258-275, August 14-18, 2005. Article (CrossRef Link)
[26] P. Wang, D. Feng and L. Zhang, "Towards attribute revocation in key-policy attribute based encryption," *CANS'11*, LNCS 7092, pp. 272-291, December 10-12, 2011. Article (CrossRef Link)
[27] Y. Cheng, Z. Wang, J. Ma, J. Wu, S. Mei and J. Ren, "Efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage," *Journal of Zhejiang University-SCIENCE C*, vol. 14, no. 2, pp. 85-97, 2013. Article (CrossRef Link)
[28] Y. Zhang, X. Chen, J. Li, H. Li and F. Li, "FDR-ABE: Attribute-based encryption with flexible and direct revocation," in *Proc. of the 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS'13)*, pp. 38-45, September 9-11, 2013. Article (CrossRef Link)
[29] B. Lynn, "The stanford pairing based crypto library," 2014. Article (CrossRef Link)

Yinghui Zhang received his B.S. (2007) and M.S. (2010) from Nanchang Hangkong University and Xidian University, both in Mathematics. He got his Ph.D degree in Cryptography from Xidian University at 2013. Currently, he works at Xi'an University of Posts and Telecommunications. His research interests are in the areas of wireless network security, cloud security and cryptography.

Xiaofeng Chen received his B.S. and M.S. on Mathematics in Northwest University, China. He got his Ph.D degree in Cryptography from Xidian University at 2003. Currently, he works at Xidian University as a professor. His research interests include applied cryptography and cloud computing security. He has published over 80 research papers in refereed international conferences and journals. His work has been cited more than 1000 times at Google Scholar. He has served as the program/general chair or program committee member in over 20 international conferences.



Jin Li received his B.S. (2002) and M.S. (2004) from Southwest University and Sun Yat-sen University, both in Mathematics. He got his Ph.D degree in information security from Sun Yat-sen University at 2007. Currently, he is a professor at Guangzhou University. His research interests include design of secure protocols in Cloud Computing and cryptographic protocols. He served as a senior research associate at Korea Advanced Institute of Technology (Korea) and Illinois Institute of Technology (U.S.A.) from 2008 to 2010, respectively. He has published more than 40 papers in international conferences and journals, including IEEE INFOCOM, IEEE Transaction on Computers, IEEE Transaction on Parallel and Distributed Computation, etc. He also served as TPC committee for many international conferences.

Hui Li received his B.Sc. degree from Fudan University in 1990, M.A.Sc. and Ph.D. degrees from Xidian University, Xi' an, China, in 1993 and 1998, respectively. He was as a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Waterloo, Ontario, Canada, in 2009. Since June 2005, he has been a professor in the school of Telecommunications Engineering, Xidian University. His research interests are in the areas of cryptography, security of cloud computing, wireless network security, information theory, and network coding. He is the co-author of two books. He served as TPC co-chair of ISPEC 2009 and IAS 2009, general co-chair of e-forensic 2010, ProvSec 2011, and ISC 2011.

Fenghua Li received the B.S. degree, M.S. degree and PhD degree in Computer Science from Xidian University in 1987, 1990 and 2009, respectively. He is a professor of Institute of Information Engineering, Chinese Academy of Sciences. And he is also a doctoral supervisor of Xidian University. His main research interests are network security and system security.