1-1-2016

Ciphertext-policy attribute based encryption supporting access policy update

Yinhao Jiang
University of Wollongong, yj971@uowmail.edu.au

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

Fuchun Guo
University of Wollongong, fuchun@uow.edu.au

Ciphertext-policy attribute based encryption supporting access policy update

Abstract

Attribute-based encryption (ABE) allows one-to-many encryption with static access control. In many occasions, the access control policy must be updated and the original encryptor might be required to re-encrypt the message, which is impractical, since the encryptor might be unavailable. Unfortunately, to date the work in ABE does not consider this issue yet, and hence this hinders the adoption of ABE in practice. In this work, we consider how to efficiently update access policies in Ciphertext-policy Attribute-based Encryption (CP-ABE) systems without re-encryption. We introduce a new notion of CP-ABE supporting access policy update that captures the functionalities of attribute addition and revocation to access policies. We formalize the security requirements for this notion, and subsequently construct two provably secure CP-ABE schemes supporting AND-gate access policy with constant-size ciphertext for user decryption. The security of our schemes are proved under the Augmented Multi-sequences of Exponents Decisional Diffie-Hellman assumption.

Keywords

supporting, encryption, access, attribute, update, policy, ciphertext

Disciplines

Engineering | Science and Technology Studies

Publication Details

Jiang, Y., Susilo, W., Mu, Y. & Guo, F. (2016). Ciphertext-policy attribute based encryption supporting access policy update. Lecture Notes in Computer Science, 10005 39-60. Nanjing, China Provable Security: 10th International Conference, ProvSec 2016 Nanjing, China, November 10-11, 2016 Proceedings

Ciphertext-Policy Attribute Based Encryption Supporting Access Policy Update

Yinhao Jiang, Willy Susilo, Yi Mu, and Fuchun Guo

Centre for Computer and Information Security Research, School of Computing and Information Technology, University of Wollongong, Australia
{yj971,wsusilo,ymu,fuchun}@uow.edu.au

Abstract. Attribute-based encryption (ABE) allows one-to-many encryption with static access control. In many occasions, the access control policy must be updated and the original encryptor might be required to re-encrypt the message, which is impractical, since the encryptor might be unavailable. Unfortunately, to date the work in ABE does not consider this issue yet, and hence this hinders the adoption of ABE in practice. In this work, we consider how to efficiently update access policies in Ciphertext-policy Attribute-based Encryption (CP-ABE) systems without re-encryption. We introduce a new notion of CP-ABE supporting access policy update that captures the functionalities of attribute addition and revocation to access policies. We formalize the security requirements for this notion, and subsequently construct two provably secure CP-ABE schemes supporting AND-gate access policy with constant-size ciphertext for user decryption. The security of our schemes are proved under the Augmented Multi-sequences of Exponents Decisional Diffie-Hellman assumption.

Keywords: Attribute-based encryption, Access policy update, Ciphertext-policy

1 Introduction

Attribute-based encryption (ABE) enforces encrypted data to be decrypted with a secure access control mechanism that the assigned attributes must satisfy the access policies associated with ciphertext and private keys. ABE has become a promising cryptographic primitive providing one-to-many encryption. The notion of ABE was put forth by Sahai and Waters [22] with the original notion called *fuzzy IBE*, and subsequently followed by many othere works. In the notion of ABE, there are two variants, namely Ciphertext-policy Attribute-based Encryption (CP-ABE) and Key-policy Attribute-based Encryption (KP-ABE), depending on the location of the access policy. In the former, the access policy is embedded in the ciphertext, whilst in the latter, the access policy is embedded in the private keys. We note that KP-ABE is less flexible than CP-ABE because in KP-ABE, once a users private key is issued the access policy is also determined, which makes the encryption more difficult as the encryptor needs to compare recipients access policies to all other users to choose a proper set of attributes for the ciphertext. In a CP-ABE system, users' keys are labelled with sets of descriptive attributes and distributed from a trusted key generation authority. Ciphertexts in the system are assigned specific access policies stating what attributes are required for its decryption. In such a system,

a ciphertext can be decrypted by a user's key only if the set of attributes associated with the user's key satisfies its access policy.

When using CP-ABE to distribute a message to a specific set of users, the encryptor simply constructs an access policy such that the receivers can only decrypt the ciphertext if they have the set of attributes that satisfy the access policy. The encryptor can just merely encrypt the message with the access policy, and then upload it to the storage server. The storage server does not need to be trusted by receivers but it functions as a proxy, which performs the task that is assigned a priori. Unfortunately, to date, there is no CP-ABE that supports changes of access policies of ciphertexts. We note that this is a highly desirable feature as situation can change from time to time, and without the ability to update the access policy, CP-ABE cannot be adopted in practice. Hence, an efficient update mechanism over access policies of ciphertexts must be enabled.

One may think that the above question can be solved trivially by requesting encryptors to re-encrypt the messages when the access policies are updated. Unfortunately, this approach is very impractical and unusable, since encryptors may not even be available during the access policy update. Alternatively, the Ciphertext-policy Attribute-based Proxy Re-encryption (CP-AB-PRE) system may be employed for access policy update as showed in Fig. 1. The CP-AB-PRE works as follows. When an access control authority has decided to update access policies of certain range of ciphertexts, he uses his own private key to generate a re-encryption key for each ciphertext from the old access policy to a new access policy, and uploads all the re-encryption keys to a proxy to modify the ciphertext. When the proxy receives the re-encryption key, it first checks if the attribute set of the owner of the re-encryption key satisfies the access policy of the ciphertext needed to be re-encrypted. If it does, then it proceeds with the re-encryption. We note that much effort have been put into developing and enhancing Attribute-based Proxy Re-encryption (AB-PRE) including CP-AB-PRE, and this solution is powerful and strong. What AB-PRE provides is an efficient mechanism of re-encryption, to wit to output the result of decrypting and encrypting to a new access policy without actually decrypting the ciphertext or knowing the plaintext, which can provide access policy update but with the restriction that the re-encryptor needs to generate valid re-encryption keys. Nevertheless, when the amount of involved ciphertexts rises, it becomes inefficient for the update initiator to generate all re-encryption keys and the upload bandwidth may also be limited, let alone it is unnecessarily that the initiator's private key should be able to decrypt all involved ciphertexts which then require some other users whose private key can decrypt to help re-encrypting. It can be seen that to update access policies for a large amount of ciphertexts will exceeds the capability of AB-PRE who specializes in re-encryption.

1.1 Our Contribution

In this work, we aim to equip the notion of Attribute-based Encryption (ABE) with access policy update. We present the notion of Ciphertext-policy Attribute Based Encryption supporting Access Policy Update (CP-ABE-APU). In our setting, the encryptor will produce encrypted data together with components used for access policy update and send them to a third party, which provides distributed storage servers and functions as access policy update proxy. This third party does not need to be trusted; it will store

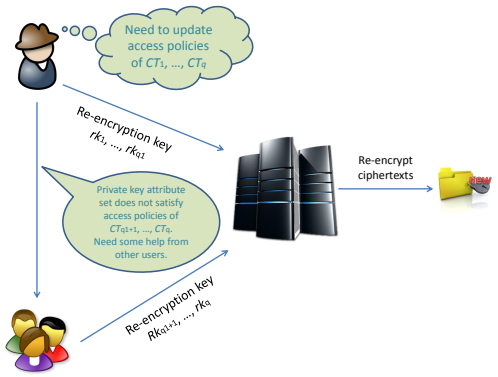

Fig. 1. An example of user updating access policies of ciphertexts employing PRE

encrypted data for users accessing and execute access policy update algorithm as requested, which does not give it the ability of decrypting any ciphertexts. We present a new security model to capture these requirements, together with two constructions supporting AND-gate access policy provably secure under augmented assumptions. In our CP-ABE-APU constructions, the ciphertext consists of 3 group elements and the components used for access policy update consist of $n - s - 1$ and t group elements for attribute addition and revocation, respectively, where in the construction for attribute addition the AND-gate access policy consists of s attributes in a ciphertext and there are n attributes in total, and in the construction for attribute revocation the maximum revocation number for a ciphertext is t (Table 1). The components for access policy update will only be stored in storage servers, which makes the ciphertext sent to users for decryption of constant size of 3. We also present the proofs of security of our constructions as well as proofs of intractability of augmented assumptions.

Table 1. Comparison between two constructions supporting access policy update

Scheme	Update operation	Attr. universe	Attr. in policy/ Max. revocation	Ciphertext for user	Ciphertext for server
Con. 1	Addition	n	s	3	$n - s + 2$
Con. 2	Revocation	n	t	3	$t + 3$

1.2 Related Work

After the notion of Attribute-based Encryption (ABE) was introduced by Sahai and Waters [22], Goyal et al. [9] proposed the first KP-ABE system, in which ciphertexts are associated with attributes, and secret keys are associated with access policies. Later, Bethencourt, Sahai and Waters [2] defined a complementary notion indicated in [22],

i.e. CP-ABE, but it is proven to be secure in generic group models. Cheung and Newport [6] presented the first CP-ABE construction whose security proof was given in the standard model, which allows the access policies to be a single AND-gate of attributes with values of positive, negative and wildcards. Goyal et al. [8] constructed a CP-ABE scheme but with large key size. Waters [25] designed efficient and expressive CP-ABE systems supporting any monotonic access structure. Attrapadung et al. [1] proposed an efficient CP-ABE for threshold access policy with constant-size ciphertexts, which adopted an algorithm Aggregate from [7] for their decryption algorithm. Later on, Waters [26] proposed the first deterministic finite automata-based functional encryption system in which access policy can be expressed by arbitrary-size regular language. Note that there are also some variants of traditional ABE in the literature, such as [21,20,13,27].

The above schemes are only selectively secure except for [2] being proven in the generic group model. Lewko et al. [11] introduced the dual system encryption technology into the ABE cryptographic setting to convert one of the CP-ABE systems proposed in [25] to achieve fully security with some loss of expressiveness. Later, Lewko and Waters [12] introduced a new method to capture full security without jeopardizing the expressiveness by employing the selective proof technique into the dual system encryption technology.

The proxy re-encryption scheme was first formalized by Blaze, Bleumer, and Strauss [3]. With the concept of ABE and PRE combined, Liang et al. [17] proposed the first CP-AB-PRE scheme based on the CP-ABE scheme [21] supporting non-monotonic access structures. Then Luo et al. [18] proposed another CP-AB-PRE scheme with multi-value positive attributes. Aside from this, Seo et al. [23] proposed a CP-AB-PRE scheme which has constant paring operation latency. Liang et al. constructed CP-AB-PRE schemes [16,15,14] proven secure in CCA security model.

Recently, Susilo et al. [24] introduced a new notion of recipient-revocable identity-based broadcast encryption scheme. In their scheme, the encryptor produces and sends ciphertexts to a proxy for broadcasting, which will also be able to revoke some identities from the original set of recipients without the knowledge the plaintext.

1.3 Roadmap

The rest of this paper is organized as follows. In Sec. 2, we present some definitions and background that will be used throughout this paper. In Sec. 3, we briefly review bilinear groups and complexity assumption that are used in this paper. We present our CP-ABE scheme that supports attribute addition in Sec. 4, together with its security analysis. Sec. 5 deals with CP-ABE that supports attribute revocation, as well as its security analysis. We presented the analysis of the intractability of the hard problem that is used to analyze our schemes in Sec. 6. The analysis is done in the generic group model. Finally, we conclude the work in Sec. 7.

2 Definitions

We first give formal definitions for the security of Ciphertext-policy Attribute Based Encryption supporting Access Policy Update. Then we give background information on pairings and complexity assumptions.

2.1 Access Structure [6]

Generally speaking, an access structure on attributes is a rule $\mathbb{A}$ that returns either 0 or 1 given an attribute set W. We say that W satisfies $\mathbb{A}$ iff $\mathbb{A}$ answers 1 on W. Access structures may be Boolean expressions, threshold trees, etc.

In this paper, we focus on access structures that consist of a single AND gate whose inputs are attributes. This is denoted $\mathbb{A} = \bigwedge_{\mathsf{at} \in S} \mathsf{at}$, where S is a subset of the attribute universe $\mathcal{P}$ and every at is an attribute in $\mathcal{P}$. Given an attribute set W, $\mathbb{A}$ answers 1 iff for all $\mathsf{at} \in S$, $\mathsf{at} \in W$. Thus, W satisfies $\mathbb{A}$ iff $S \subseteq W$. Since AND-gates are sufficient in many application scenarios, our approach retains significant potential.

2.2 CP-ABE supporting Access Policy Update Definition

A ciphertext-policy attribute-based encryption system supporting attribute addition consists of five algorithms: Setup, Encrypt, KeyGen, Update and Decrypt.

$\mathsf{Setup}(1^\lambda, \mathcal{P})$**.** The setup algorithm takes input the attribute universe $\mathcal{P}$ as well as the implicit security parameter. It outputs the public parameters params and a master secret key msk.

$\mathsf{Enc}(\mathsf{params}, M, \mathbb{A})$**.** The encryption algorithm takes in the public parameters params, the message M, and an access structure $\mathbb{A}$ over the universe of attributes. It will output a ciphertext CT such that only users whose private keys associated with attribute sets which satisfy the access structure $\mathbb{A}$ can decrypt M. We assume that the ciphertext implicitly contains $\mathbb{A}$.

$\mathsf{KeyGen}(\mathsf{msk}, W)$**.** The key generation algorithm takes as input the master secret msk and a set of attributes W. It outputs a private key sk associated with W.

$\mathsf{Update}(\mathsf{params}, CT, opt, \mathcal{U})$**.** The addition algorithm takes as input the public parameters params, a ciphertext CT for an access policy $\mathbb{A} = \bigwedge_{\mathsf{at} \in S} \mathsf{at}$, an operation indicator $opt = \mathsf{Add}$ or Revoke and a set of attributes $\mathcal{U}$ with $\mathcal{U} \cap S = \emptyset$ if $opt = \mathsf{Add}$ or $\mathcal{U} \subset S$ if $opt = \mathsf{Revoke}$. It outputs a new ciphertext CT' for the new access policy $\mathbb{A}' = \bigwedge_{\mathsf{at} \in S \cup \mathcal{U}}$ or $\bigwedge_{\mathsf{at} \in S \setminus \mathcal{U}}$ according to opt.

$\mathsf{Dec}(\mathsf{params}, CT, sk)$**.** The decryption algorithm takes as input the public parameters PK, a ciphertext CT for an access structure $\mathbb{A}$, and a private key sk associated with a set of attributes W. If the attribute set W satisfies the access structure $\mathbb{A}$ then the algorithm will decrypt the ciphertext and return a message M.

Selective CPA Security Model for CP-ABE supporting Access policy Update. We now give the security definition for CP-ABE system – Indistinguishability under selective chosen plaintext attacks ($\mathsf{IND\text{-}sCPA}$ security, for short). This is described by a security game between a challenger and an adversary for a security parameter $\lambda \in \mathbb{N}$. The game proceeds as follows:

Init The challenger defines an attribute universe $\mathcal{P}$ of size n and gives it to the adversary $\mathcal{A}$. $\mathcal{A}$ chooses a challenge access structure $\mathbb{A}^*$ of one attribute set $S \subset \mathcal{P}$ with $s = |S|$, and gives it to the challenger.

Setup The challenger runs the Setup algorithm and gives the public parameters params to the adversary.

Phase 1 The adversary queries the challenger for private keys corresponding to sets of attributes $W_1, \ldots, W_{q_1}$ with the restriction that none of these satisfies the access policy $\mathbb{A}^*$.

Challenge The adversary declares two equal length messages M_0 and M_1 as well as a attribute set $\mathcal{U}^*$ with $t = |\mathcal{U}^*|$ and $\mathcal{U}^* \subset S$ or $\mathcal{U}^* \cap S = \emptyset$ according to "*opt*" $= \mathsf{Add}$ or "*opt*"$= \mathsf{Revoke}$ respectively. The challenger flips a random coin $\beta \in \{0, 1\}$, and encrypts M_β with $\mathbb{A}' = \bigwedge_{\mathsf{at} \in S \setminus \mathcal{U}^*} \mathsf{at}$ for "*opt*"$= \mathsf{Add}$ or $\mathbb{A}' = \bigwedge_{\mathsf{at} \in S \cup \mathcal{U}^*} \mathsf{at}$ for "*opt*"$= \mathsf{Revoke}$, producing $CT^* = \mathsf{Enc}(\mathsf{params}, \mathbb{A}^*, M_\beta)$. It gives CT^* to the adversary if $\mathcal{U}^* = \emptyset$, otherwise $CT' = \mathsf{Update}(\mathsf{params}, CT^*, opt, \mathcal{U}^*)$.

Phase 2 The adversary queries the challenger for private keys corresponding to sets of attributes $W_{q_1+1}, \ldots, W_q$ with the same restriction that none of these satisfies the access policy $\mathbb{A}^*$.

Guess The adversary outputs a guess β' for β.

The advantage of an adversary in winning this game is defined to be

$$\mathsf{Adv}^{\mathsf{IND\text{-}sCPA}}_{\mathcal{A},\mathsf{CP\text{-}ABE\text{-}AA}} = |\Pr[\beta' = \beta] - \frac{1}{2}|.$$

Definition 1. *A ciphertext-policy attribute-based encryption system supporting access policy update is selective chosen-plaintext attack secure if all polynomial time adversaries have at most a negligible advantage in this security game.*

It is worth noticing that our newly defined security model has two different types of attackers considered.

1. When $\mathcal{U}^* = \emptyset$, the challenge ciphertext CT^* is the direct result of encryption algorithm without any involvement of access policy update algorithm. It can be seen that this is essentially the property of $\mathsf{IND\text{-}sCPA}$ security for CP-ABE schemes that an adversary who does not hold a private key associated with a set of attributes satisfying the challenge access policy cannot distinguish which submitted message was encrypted as the challenge ciphertext.
2. When $\mathcal{U}^* \neq \emptyset$, the challenge ciphertext CT' is the result of updating $\mathcal{U}^*$ from $\mathbb{A}'$ of the ciphertext of encrypted M_β. It can be seen that in this situation it prevents the type of attackers who obtain private keys associated with any attributes satisfying access policy before update from learning anything about the plaintext.

3 Pairings and Complexity Assumption

Our construction will make use of groups with bilinear maps [5], and two new computational assumptions, that fit into the General Diffie-Hellman Exponent framework proposed by Boneh, Boyen and Goh [4].

3.1 Bilinear Maps

Let $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ be three cyclic groups of prime order p. A bilinear map $e(\cdot,\cdot)$ is a map $\mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ such that for any generators $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_p$, the following three conditions hold:

1. *Bilinearity* $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$.
2. *Non-degeneracy* $e(g_1, g_2) \neq 1$.
3. *Computability* There exists efficient algorithms to compute all group operations as well as the bilinear map $e(\cdot,\cdot)$.

A bilinear map group system is a tuple $\mathbb{S} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e(\cdot,\cdot))$, composed of objects as described above.

In our construction, an arbitrary bilinear map group system is adopted, without any specific additional property. In particular, it does not require $\mathbb{G}_1$ and $\mathbb{G}_2$ to be distinct or equal. Neither does it need an efficient isomorphism from $\mathbb{G}_1$ to $\mathbb{G}_2$, and vice versa.

3.2 Complexity Assumption

The security of our schemes are reduced to the hardness of a problem, which we called the *augmented multi-sequence of exponents decisional Diffie-Hellman problem*. The problems are modified from the (l, m, t)-aMSE-DDH problem defined in [10], of which the generic complexity is covered by the general Diffie-Hellman exponent theorem due to Boneh, Boyen and Goh [4], as the problem lies in the scope of their framework.

First we introduce the assumption which our CP-ABE-AA scheme is reduced to. Let $\mathbb{S} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e(\cdot,\cdot))$ be a bilinear map group system. Let g_0 be a generator of $\mathbb{G}_1$ and h_0 be a generator of $\mathbb{G}_2$. Let n, s be two integers. The first (n, s)-*augmented multi-sequence of exponents decisional Diffie-Hellman* ((n, s)-aMSE-DDH$_A$) *problem* related to $\mathbb{S}$ is as follows:

Input The vector $\overrightarrow{x}_n = (x_1, \ldots, x_n)$ defines the coprime polynomials, of which the components are pairwise distinct elements of $\mathbb{Z}_p$,

$$f(X) = \prod_{i=1}^{n-s}(X + x_i), \quad g(X) = \prod_{i=n-s+1}^{n}(X + x_i),$$

the values

$$\begin{cases} g_0, g_0^{\gamma}, \ldots, g_0^{\gamma^{n-2}}, g_0^{\kappa\cdot\gamma\cdot f(\gamma)}, & (1.1) \\ g_0^{\alpha}, g_0^{\alpha\cdot\gamma}, \ldots, g_0^{\alpha\cdot\gamma^{n-s+1}}, & (1.2) \\ g_0^{\omega\cdot\gamma}, \ldots, g_0^{\omega\cdot\gamma^{n-1}}, & (1.3) \\ h_0, h_0^{\gamma}, \ldots, h_0^{\gamma^{s-2}}, & (1.4) \\ h_0^{\kappa\cdot g(\gamma)}, h_0^{\kappa\cdot\gamma\cdot g(\gamma)}, \ldots, h_0^{\kappa\cdot\gamma^{n-s}\cdot g(\gamma)}, & (1.5) \\ h_0^{\alpha}, h_0^{\alpha\cdot\gamma}, \ldots, h_0^{\alpha\cdot\gamma^{n}}, & (1.6) \\ h_0^{\omega}, h_0^{\omega\cdot\gamma}, \ldots, h_0^{\omega\cdot\gamma^{s-1}}, & (1.7) \end{cases}$$

where κ,ω, α, γ are unknown random elements of $\mathbb{Z}_p$, element $T_b = e(g_0, h_0)^{\kappa\cdot f(\gamma)} \in \mathbb{G}_T$ and a random group element $T_{1-b} \in \mathbb{G}_T$ while b is a fair coin.

Output a bit b'. The problem is correctly solved if the output is $b' = b$.

The following statement is a corollary of Theorem A.2 in [4]. It provides an intractability bound in the generic model, but in groups equipped with pairings. We emphasize on the fact that, whereas the assumption has several parameters, it is non-interactive, and thus easily falsifiable [19].

Corollary 1 (Generic Security). *For any probabilistic algorithm $\mathcal{B}$ that makes at most q_G queries to the oracles performing group operations in $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ and the bilinear map $e(\cdot,\cdot)$, its advantage in solving $(n,\ s)$-$\mathsf{aMSE\text{-}DDH}_A$ problem is bounded as*

$$\mathsf{Adv}_{\mathcal{B}}^{(n,s)\text{-}\mathsf{aMSE\text{-}DDH}_A}(\lambda) \leq \frac{(q_G + 5n + 3)^2 \cdot d}{2p}$$

where $d = 2n$.

Second, we introduce the assumption for our CP-ABE-AR scheme. Let $\mathbb{S} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e(\cdot,\cdot))$ be a bilinear map group system. Let g_0 be a generator of $\mathbb{G}_1$ and h_0 be a generator of $\mathbb{G}_2$. Let n, s be two integers. The second (n, s)*-augmented multi-sequence of exponents decisional Diffie-Hellman* $((n, s)$-$\mathsf{aMSE\text{-}DDH}_B)$ *problem* related to $\mathbb{S}$ is as follows:

Input The vector $\overrightarrow{x}_n = (x_1, \ldots, x_n)$ defines the coprime polynomials, of which the components are pairwise distinct elements of $\mathbb{Z}_p$,

$$f(X) = \prod_{i=1}^{n-s}(X + x_i), \quad g(X) = \prod_{i=n-s+1}^{n}(X + x_i),$$

the values

$$\begin{cases} g_0, g_0^{\gamma}, \ldots, g_0^{\gamma^{n-2}}, g_0^{\kappa\cdot\gamma\cdot f(\gamma)}, & (2.1) \\ g_0^{\alpha}, g_0^{\alpha\cdot\gamma}, \ldots, g_0^{\alpha\cdot\gamma^{2n-s}}, & (2.2) \\ g_0^{\omega\cdot\gamma}, \ldots, g_0^{\omega\cdot\gamma^{n-1}}, & (2.3) \\ h_0, h_0^{\gamma}, \ldots, h_0^{\gamma^{s-2}}, & (2.4) \\ h_0^{\kappa\cdot g(\gamma)}, & (2.5) \\ h_0^{\alpha}, h_0^{\alpha\cdot\gamma}, \ldots, h_0^{\alpha\cdot\gamma^{n}}, & (2.6) \\ h_0^{\omega}, h_0^{\omega\cdot\gamma}, \ldots, h_0^{\omega\cdot\gamma^{s-1}}, & (2.7) \end{cases}$$

where $\kappa, \omega, \alpha, \gamma$ are unknown random elements of $\mathbb{Z}_p$, element $T_b = e(g_0, h_0)^{\kappa\cdot f(\gamma)} \in \mathbb{G}_T$ and a random group element $T_{1-b} \in \mathbb{G}_T$ while b is a fair coin.

Output a bit b'. The problem is correctly solved if the output is $b' = b$.

Corollary 2 (Generic Security). *For any probabilistic algorithm $\mathcal{B}$ that makes at most q_G queries to the oracles performing group operations in $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ and the bilinear map $e(\cdot,\cdot)$, its advantage in solving $(n,\ s)$-$\mathsf{aMSE\text{-}DDH}_B$ problem is bounded as*

$$\mathsf{Adv}_{\mathcal{B}}^{(n,s)\text{-}\mathsf{aMSE\text{-}DDH}_B}(\lambda) \leq \frac{(q_G + 5n + s + 4)^2 \cdot d}{2p}$$

where $d = 2(2n - s)$.

4 CP-ABE Supporting Attribute Addition Construction

In this section, we shall present our ciphertext-policy attribute-based encryption scheme that supports access policy update with operation indicator $opt = \mathsf{Add}$.

Before presenting the description of our scheme, we introduce the adopted algorithm Aggregate of [7] for the decryption process. This algorithm is given for group elements in $\mathbb{G}_T$ [7], but it can be seen that it works in any group of prime order.

$\mathsf{Aggregate}(\{g^{\frac{r}{\gamma+x_i}}, x_i\}_{1\leq i\leq n})$ The algorithm takes in values $\{g^{\frac{r}{\gamma+x_i}}, x_i\}_{1\leq i\leq n}$, where $g^{\frac{r}{\gamma+x_i}} \in \mathbb{G}_1$, $r, \gamma \in \mathbb{Z}_p$ are unknown and x_i's are pairwise distinct. It outputs the value $\mathsf{Aggregate}(\{g^{\frac{r}{\gamma+x_i}}, x_i\}_{1\leq i\leq n}) = g^{\frac{r}{\prod_{i=1}^{n}(\gamma+x_i)}} \in \mathbb{G}_1$.

4.1 Description

$\mathsf{Setup}(1^\lambda, \mathcal{P})$ The PKG chooses a suitable encoding τ sending each attribute in $\mathcal{P}$ onto (different) elements $\tau(\mathsf{at}) = \delta \in \mathbb{Z}_p$. It also chooses a bilinear group system $\mathbb{S} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e(\cdot,\cdot))$. It picks at random two generators g of $\mathbb{G}_1$ and h of $\mathbb{G}_2$. Then, the PKG picks at random $\alpha, \gamma \in \mathbb{Z}_p$ and sets $u = g^{\alpha\gamma}$, and $v = e(g^\alpha, h)$. The master secret key is then $\mathsf{msk} = (g, \alpha, \gamma)$ and the public parameters are

$$\mathsf{params} = \left(\mathcal{P}, n, u, v, h, \{h^{\alpha\gamma^i}\}_{i=0,\ldots,n}, \tau\right).$$

$\mathsf{KeyGen}(\mathsf{params}, W, \mathsf{msk})$ Given any subset $W \subset \mathcal{P}$ of attributes, the PKG picks $r \in \mathbb{Z}_p$ at random, computes $sk_W = \left(\{g^{\frac{r}{\gamma+\tau(\mathsf{at}_i)}}\}_{\mathsf{at}_i\in W}, h^{\frac{r-1}{\gamma}}\right)$.

$\mathsf{Enc}(\mathsf{params}, M, \mathbb{A})$ Given an AND-gate access structure of a set of attributes $S \subset \mathcal{P}$ with $s = |S|$, and a message $M \in \mathbb{G}_T$, the sender picks at random $\kappa \in \mathbb{Z}_p$ and computes

$$\begin{cases} E_0 = h^{\kappa\cdot\alpha\cdot\prod_{\mathsf{at}\in S}(\gamma+\tau(\mathsf{at}))}, E_1 = E_0^\gamma, \ldots, E_{n-s} = E_{n-s-1}^\gamma \\ C_1 = u^{-\kappa}, \\ C_M = v^\kappa \cdot M \end{cases}$$

The ciphertext sent from its encryptor to the storage server is then $CT_{\mathsf{server}} = (E_0, \ldots, E_{n-s}, C_1, C_M)$ while the part of $CT = (E_0, C_1, C_M)$ will be accessed by users for decryption.

$\mathsf{Update}(\mathsf{params}, CT, \text{“add”}, \mathcal{U})$ Given a ciphertext CT with an AND-gate access structure of attribute set S and a set of attributes $\mathcal{U} = \{\mathsf{at}'_1, \ldots, \mathsf{at}'_t\}$ with $t = |\mathcal{U}|$ and $\mathcal{U} \cap S = \emptyset$, the proxy adds attributes in $\mathcal{U}$ to the AND-gate access structure of the ciphtertext CT as follows.

Let $F(x)$ be the polynomial in x as $F(x) = \prod_{\mathsf{at}'\in\mathcal{U}}(x + \tau(\mathsf{at}')) = f_t x^t + f_{t-1}x^{t-1} + \cdots + f_0$.

Compute $E'_0 = E_0^{F(\gamma)} = \prod_{i=0}^{t} E_i^{f_i}$. Then new ciphertext is then $CT' = (E'_0, C_1, C_M)$ with its AND-gate access structure $\mathbb{A}'$ of attribute set $S \cup \mathcal{U}$.

$\mathsf{Dec}(\mathsf{params}, CT, sk_W)$ Any user with a set of attributes W such that $W \models \mathbb{A}$ can use the private key to decrypt the ciphertext.
First, the user computes $e(g,h)^{\kappa\cdot\alpha\cdot r}$ as follows. The user computes

$$\mathsf{Aggregate}(\{g^{\frac{r}{\gamma+\tau(\mathsf{at}_i)}}, \tau(\mathsf{at}_i)\}_{\mathsf{at}_i\in S}) = g^{\frac{r}{\prod_{\mathsf{at}_i\in S_1}\gamma+\tau(\mathsf{at}_i)}}.$$

With the output the user then computes $e(g,h)^{\kappa\cdot\alpha\cdot r} = e(g^{\frac{r}{\prod_{\mathsf{at}_i\in S_1}\gamma+\tau(\mathsf{at}_i)}}, E_0)$. After that, the user computes $e(g,h)^{\kappa\cdot\alpha} = e(C_1, h^{\frac{r-1}{\gamma}})\cdot e(g,h)^{\kappa\cdot\alpha\cdot r}$. Finally, the user recovers the message $M = \frac{C_M}{e(g,h)^{\kappa\cdot\alpha}}$.

4.2 Security Analysis

In this section, we are going to prove that our CP-ABE-AA scheme is secure against selective chosen-ciphertext attack, assuming that the (n,s)-$\mathsf{aMSE\text{-}DDH}_A$ problem is hard to solve.

Theorem 1. *Let λ be an integer. For any adversary $\mathcal{A}$ against the* $\mathsf{IND\text{-}sCPA}$ *security of our CP-ABE-AA encryption scheme $\mathcal{S}_{AA}$, for an attribute universe $\mathcal{P}$ of size n, and a challenge set$S$ with $s = |S|$, there exists an algorithm B of the (n,s)-$\mathsf{aMSE\text{-}DDH}_A$ problem, such that*

$$\mathsf{Adv}_{\mathcal{B}}^{(n,s)\text{-}\mathsf{aMSE\text{-}DDH}_A}(\lambda) \geq \mathsf{Adv}_{\mathcal{A},\mathcal{S}_{AA}}^{\mathsf{IND\text{-}sCPA}}(\lambda).$$

Proof. We now give the details of the simulation. From now on, we will denote by W_S the subset $W \cap S$.

Init $\mathcal{B}$ defines an attribute universe $\mathcal{P} = \{\mathsf{at}_1, \ldots, \mathsf{at}_n\}$ of cardinal n. $\mathcal{A}$ gives $\mathcal{B}$ the challenge access structure $\mathbb{A}^*$ defined by an AND-gate policy $\bigwedge_{\mathsf{at}\in S}\mathsf{at}$ where $S \subset \mathcal{P}$ of respective cardinal s. Here we assume $S = \{\mathsf{at}_{n-s+1}, \ldots, \mathsf{at}_n\}$.

Setup The algorithm $\mathcal{B}$ defines $g := g_0^{f(\gamma)}$, $h := h_0$. $\mathcal{B}$ then can compute
– the value $u = g^{\alpha\gamma} = g_0^{\alpha\gamma\cdot f(\gamma)}$ with line (1.2) of its input values, since the exponent $\alpha\cdot\gamma\cdot f(\gamma)$ is a linear combination of $\{\alpha, \alpha\cdot\gamma, \ldots, \alpha\cdot\gamma^{n-s+1}\}$ and $\mathcal{B}$ knows the coefficients of the exponent polynomial.
– the value $v = e(g,h)^{\alpha} = e(g_0^{\alpha\cdot f(\gamma)}, h_0)$ with line (1.2) and line (1.4).
– elements in $\{h^{\alpha\gamma^i} = h_0^{\alpha\cdot\gamma^i}\}_{i=0,\ldots,n}$ with line (1.6).

– The encoding τ is defined as $\tau(\mathsf{at}_i) = x_i$ for $i = 1, \ldots, n$. It can be seen that the encodings of the first $n-s$ elements are the opposite of the roots of $f(X)$, the encodings of the attributes in S are the opposite of roots of $g(X)$.

Finally, $\mathcal{B}$ sends to $\mathcal{A}$ the simulated public parameters: $\Big(u, v, h, \{h^{\alpha\gamma^i}\}_{i=0,\ldots,n}, \tau\Big)$.

Phase 1 The adversary $\mathcal{A}$ makes private key queries. To respond to a query on attribute set $W \subset \mathcal{P}$, where $W \not\models \mathbb{A}^*$, the algorithm $\mathcal{B}$ must produce a tuple of the form $\Big(\{g^{\frac{r}{\gamma+\tau(\mathsf{at})}}\}_{\mathsf{at}\in W}, h^{\frac{r-1}{\gamma}}\Big)$.

Observe that since $W \not\models \mathbb{A}^*$ all allowed queries must satisfy $|W_S| < s$. $\mathcal{B}$ defines the polynomial $Q_{W_S}(X) = \begin{cases} 1 & |W_S| = 0 \\ \lambda_i \cdot \prod_{\mathsf{at} \in W_S}(X + \tau(\mathsf{at})) & |W_S| > 0 \end{cases}$, where $\lambda = \left(\prod_{A \in \omega_S} \tau(\mathsf{at})\right)^{-1}$, and simulates a private key for W as follows:

$\mathcal{B}$ picks at random y_W in $\mathbb{Z}_p$, and defines $r := (1 + \omega y_W \gamma) Q_{W_S}(\gamma)$. $\mathcal{B}$ then computes the elements for sk_W:

– For any attribute $\mathsf{at} \in W$, $g^{\frac{r}{\gamma+\tau(\mathsf{at})}} = g_0^{\omega\gamma y_W \cdot \frac{f(\gamma)Q_{W_S}(\gamma)}{\gamma+\tau(\mathsf{at})}} \cdot g_0^{\frac{f(\gamma)Q_{W_S}(\gamma)}{\gamma+\tau(\mathsf{at})}}$. Since an attribute $\mathsf{at} \in W$ can be in W_S or $\mathcal{P} \setminus S$, $(\gamma + \tau(\mathsf{at}))|f(\gamma)Q_{W_S}(\gamma)$. The first factor can be computed with line (1.3) as its exponent is a polynomial in γ of degree at most $n-1$, and the second factor can be computed with line (1.1) as its exponent is a polynomial in γ of degree at most $n-2$.

– The value $h^{\frac{r-1}{\gamma}} = h_0^{\omega y_W Q_{W_S}(\gamma)} \cdot h_0^{\frac{Q_{W_S}(\gamma)-1}{\gamma}}$, where the first factor can be computed from line (1.7) and the second factor can be computed from line (1.4), since $Q_{W_S}(\gamma)$ is a polynomial with independent term 1 by its definition, thus $\frac{Q_{W_S}(\gamma)-1}{\gamma}$ is a linear combination of $\{1, \gamma, \ldots, \gamma^{s-2}\}$.

Challenge Once $\mathcal{A}$ sends to $\mathcal{B}$ the two messages M_0 and M_1 as well as an update attribute set $\mathcal{U}^*$, $\mathcal{B}$ flips a coin $\beta \in \{0, 1\}$, and sets $C'_M = T_0 \cdot M_\beta$. To simulate the rest of the challenge ciphertext, $\mathcal{B}$ implicitly defines the randomness for the encryption as $\kappa^* = \kappa/\alpha$, and sets $E'_0 = h^{\kappa^* \alpha \cdot g(\gamma)} = h_0^{\kappa \cdot g(\gamma)}$ which is given in line (1.5) as well as $E'_1, \ldots, E'_{n-s}$. To complete the ciphertext, B computes $C'_1 = u^{-\kappa^*} = g_0^{-\kappa\gamma f(\gamma)}$ from line (1.1). $\mathcal{B}$ gives $\mathcal{A}$ the challenge ciphertext $CT^* = (E'_0, E'_1, \ldots, E'_{n-s}, C'_1, C'_M)$.
Phase 2 After the challenge step $\mathcal{A}$ may make other key extraction queries, which are answered as before.
Guess $\mathcal{A}$ outputs a β'. If $\beta' = \beta$, $\mathcal{B}$ outputs 0; otherwise $\mathcal{B}$ outputs 1.

Probability Analysis:
Let $\mathcal{I} = (\overrightarrow{x}_n, \gamma, \kappa, \omega, \alpha, T_b, T_{1-b})$ be the input of the algorithm $\mathcal{B}$ and the adversary $\mathcal{A}$ break our CP-ABE scheme with advantage $\mathsf{Adv}^{\mathsf{IND\text{-}sCPA}}_{\mathcal{A},\mathcal{S}_{AA}}(\lambda)$. Below we analyse the simulation in two cases.

Case 1 ($\mathcal{U}^* = \emptyset$). Let $\kappa = \kappa^* \cdot \alpha$. One can verify that in this case, $E'_0 = h_0^{\kappa \cdot g(\gamma)} = h^{\kappa^* \cdot \alpha \cdot \prod_{\mathsf{at} \in S}(\gamma+\tau(\mathsf{at}))}$ and $C'_1 = g_0^{-\kappa \cdot \gamma \cdot f(\gamma)} = g_0^{-\kappa^* \cdot \alpha \cdot \gamma \cdot f(\gamma)} = u^{-\kappa^*}$. As for the C'_M, we also note that if $b = 0$, $T_0 = e(g0, h0)^{\kappa f(\gamma)}$, then $C'_M = e(g_0, h_0)^{\kappa f(\gamma)} \cdot M_\beta = e(g^\alpha, h)^{\kappa^*} \cdot M_\beta = v^{\kappa^*} \cdot M_\beta$. Therefore, the simulation of $\mathcal{B}$ is perfect, and the adversary $\mathcal{A}$ will guess the bit β with its advantage. Hence, if $b = 0$ we have

$$|\Pr[\mathcal{B}(\mathcal{I}) = 0 | b = 0] - \frac{1}{2}| = \mathsf{Adv}^{\mathsf{IND\text{-}sCPA}}_{\mathcal{A}}(\lambda).$$

Else, if $b = 1$ and T_0 is uniformly random in $\mathbb{G}_T$, C'_M is uniformly random and independent in $\mathbb{G}_T$, and the value of β is independent from $\mathcal{A}$'s view as well,

$$\Pr[\mathcal{B}(\mathcal{I}) = 0 | b = 1] = \frac{1}{2}.$$

Thus, we have the advantage of $\mathcal{B}$ in solving the (n, s)-aMSE-DDH$_B$ problem in Case 1 is

$$\begin{aligned}\mathsf{Adv}_{\mathcal{B}}^{(\mathsf{n},\mathsf{s})-\mathsf{aMSE_B\text{-}DDH}}(\lambda) &= |\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 0] - \Pr[\mathcal{B}(\mathcal{I}) = 0|b = 1]| \\ &\geq \mathsf{Adv}_{\mathcal{A}}^{\mathsf{IND\text{-}sCPAwAR}}(\lambda).\end{aligned}$$

Case 2 ($\mathcal{U}^* = \{\mathsf{at}'_1, \mathsf{at}'_2, \ldots, \mathsf{at}'_t\} \neq \emptyset$). In this case, we first show that how a challenge ciphertext should be produced in a real game. Formally, the correct procedures are as follows.

Let $S' = S \setminus \mathcal{U}^*$. The encryption algorithm $\mathsf{Enc}(\mathsf{params}, \mathbb{A}' = \bigwedge_{\mathsf{at}\in S'} \mathsf{at}, M_\beta)$ is run to get CT^*. More precisely, it picks a randomness $\kappa' \in \mathbb{Z}_p$ and computes,

$$\begin{aligned}CT^* &= (E_0^*, E_1^*, \cdots, E_{n-s+t}^*, C_1^*, C_M^*) \\ &= (h^{\kappa'\cdot\alpha\cdot\prod_{\mathsf{at}\in S'}(\gamma+\tau(\mathsf{at}))}, \ldots, h^{\kappa'\cdot\alpha\cdot\gamma^{n-s+t}\cdot\prod_{\mathsf{at}\in S'}(\gamma+\tau(\mathsf{at}))}, u^{-\kappa'}, v^{\kappa'}\cdot M).\end{aligned}$$

The Addition algorithm $\mathsf{Add}(\mathsf{params}, CT^*, \mathcal{U}^*)$ is run to add the attribute set $\mathcal{U}^*$ to the access policy of the ciphertext CT^*. It processes as follows.

Let $F^*(x)$ be the polynomial in x as $F^*(x) = \prod_{\mathsf{at}'\in\mathcal{U}^*}(x + \tau(\mathsf{at}')) = f_t^* x^t + f_{t-1}^* x^{t-1} + \cdots + f_0^*$.

Compute ${E'_0}^* = (E_0^*)^{F^*(\gamma)} = \prod_{i=0}^{t}(E_i^*)^{f_i^*}$.

Finally, the challenge ciphertext in a real game is produced $CT' = ({E'_0}^*, C_1^*, C_M^*)$.

Now we assume that the randomness κ' used in producing CT^* is defined as $\kappa'\cdot\alpha = \kappa$. The challenge ciphertext CT' turns out to be as follows,

$$C_M^* = M_\beta \cdot v^{\kappa'} = M_\beta \cdot v^{\frac{\kappa}{\alpha}},$$

$${E'_0}^* = h^{\kappa'\cdot\alpha\cdot\prod_{\mathsf{at}\in S}(\gamma+\tau(\mathsf{at}))} = h^{\kappa\cdot\prod_{\mathsf{at}\in S}(\gamma+\tau(\mathsf{at}))} = h_0^{\kappa\cdot g(\gamma)},$$

$$C_1^* = u^{-\kappa'} = g_0^{\kappa\cdot\gamma\cdot f(\gamma)}.$$

It can be seen that if $b = 0$, $T_0 = e(g_0, h_0)^{\kappa\cdot f(\gamma)}$, the challenge ciphertext in a real game is exactly the same as the simulated challenge ciphertext. The simulated game would be a perfect simulation if it can be proved that the setting of κ' is indistinguishable from a real random value from the view of $\mathcal{A}$. It will suffice as κ is random to $\mathcal{A}$. Thus, if $b = 0$ we have

$$|\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 0] - \frac{1}{2}| = \mathsf{Adv}_{\mathcal{A},\mathcal{S}_{AA}}^{\mathsf{IND\text{-}sCPA}}(\lambda).$$

On the other hand, if $b = 1$ and T_0 is a random element from $\mathbb{G}_T$, C_M^* is random and independent from the view of $\mathcal{A}$,

$$\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 1] = \frac{1}{2}.$$

Thus, we have the advantage of $\mathcal{B}$ in solving the (n, s)-aMSE-DDH$_B$ problem in Case 2 is

$$\begin{aligned}\mathsf{Adv}_{\mathcal{B}}^{(\mathsf{n},\mathsf{s})-\mathsf{aMSE_B\text{-}DDH}}(\lambda) &= |\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 0] - \Pr[\mathcal{B}(\mathcal{I}) = 0|b = 1]| \\ &\geq \mathsf{Adv}_{\mathcal{A},\mathcal{S}_{AA}}^{\mathsf{IND\text{-}sCPAwAR}}(\lambda).\end{aligned}$$

This completes the proof. □

5 CP-ABE Supporting Attribute Revocation Construction

In this section, we shall present our ciphertext-policy attribute-based encryption scheme that supports access policy update with operation indicator $opt = \mathsf{Revoke}$.

5.1 Description

$\mathsf{Setup}(1^\lambda, \mathcal{P})$ The PKG selects a suitable encoding τ sending each attribute in $\mathcal{P}$ onto different elements $\tau(\mathsf{at}) = \delta \in \mathbb{Z}_p$. It also chooses a bilinear group system $\mathbb{S} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e(\cdot,\cdot))$. It picks at random two generators g of $\mathbb{G}_1$ and h of $\mathbb{G}_2$. Then, the PKG picks at random $\alpha, \gamma \in \mathbb{Z}_p$ and sets $\{u_i = g^{\alpha\gamma^i}\}_{i=1\ldots n}$, and $v = e(g^\alpha, h)$.
The master secret key is then $\mathsf{msk} = (g, \alpha, \gamma)$ and the public parameters are

$$\mathsf{params} = \left(\mathcal{P}, n, \{u_i\}_{i=1,\ldots,n}, v, h, \{h^{\alpha\gamma^i}\}_{i=0,\ldots,n}, \tau\}\right).$$

$\mathsf{KeyGen}(\mathsf{params}, W, \mathsf{msk})$ Given any subset $W \subset \mathcal{P}$ of attributes, the PKG picks $r \in \mathbb{Z}_p$ at random, computes $sk_W = \left(\{g^{\frac{r}{\gamma+\tau(\mathsf{at}_i)}}\}_{\mathsf{at}_i \in W}, h^{\frac{r-1}{\gamma}}\right)$.

$\mathsf{Enc}(\mathsf{params}, M, \mathbb{A}, l)$ Given an AND-gate access structure of a set of attributes $S \subset \mathcal{P}$ with $s = |S|$, a message $M \in \mathbb{G}_T$ and an extra input which is a maximum revocation number $l \leq s$, the sender picks at random $\kappa \in \mathbb{Z}_p$ and computes

$$\begin{cases} E_0 = h^{\kappa\cdot\alpha\cdot\prod_{\mathsf{at}\in S}(\gamma+\tau(\mathsf{at}))} \\ C_1 = u_1^{-\kappa}, \ldots, C_{l+1} = u_{l+1}^{-\kappa}, \\ C_M = v^\kappa \cdot M \end{cases}$$

The ciphertext sent from its encryptor to the storage server is then $CT_{\mathsf{server}} = (E_0, C_1, \ldots, C_{t+1}, C_M)$ while the part of $CT = (E_0, C_1, C_M)$ will be access by users for decryption.

$\mathsf{Update}(\mathsf{params}, CT, \text{“revoke”}, \mathcal{U})$ Given a ciphertext $CT = (E_0, C_1, \ldots, C_{t+1}, C_M)$ for an AND-gate access structure $\mathbb{A} = \wedge_{\mathsf{at}\in S}\mathsf{at}$, a revocation attribute set $\mathcal{U} = \{\mathsf{at}'_1, \ldots, \mathsf{at}'_t\} \subseteq S$ with $t \leq l$ and the public parameters params, the revocation update algorithm works as follows.
Let $F(x)$ be the polynomial in x as

$$F(x) = \frac{1}{\prod_{\mathsf{at}'\in\mathcal{U}}\tau(\mathsf{at}')}\prod_{\mathsf{at}'\in\mathcal{U}}(x+\tau(\mathsf{at}')) = f_t x^t + f_{t-1}x^{t-1} + \cdots + f_0.$$

Compute

- $C'_M = C_M \cdot e(\prod_{i=1}^t C_i^{-f_i}, h) = M \cdot e(g^{\kappa\cdot\alpha\cdot\sum_{i=0}^t f_i\gamma^i}, h) = M \cdot v^{\kappa\cdot F(\gamma)}$,
- $E'_0 = E_0^{\frac{1}{\prod_{\mathsf{at}'\in\mathcal{U}}\tau(\mathsf{at}')}} = h^{\kappa\cdot\alpha\cdot\prod_{\mathsf{at}\in S\setminus\mathcal{U}}(\gamma+\tau(\mathsf{at}))\cdot F(\gamma)}$,
- $C'_1 = \prod_{i=1}^{t+1} C_i^{f_{i-1}} = g^{-\kappa\cdot\alpha\cdot\gamma\cdot F(\gamma)} = u_1^{-\kappa\cdot F(\gamma)}$.

The new ciphertext is then $CT = (E'_0, C'_1, C'_M)$ with new randomness $\kappa \cdot F(\gamma)$.

$\mathsf{Dec}(\mathsf{params}, CT, sk_W)$ Any user with a set of attributes W such that $W \models \mathbb{A}$ can use the private key to decrypt the ciphertext.
First, the user computes $e(g,h)^{\kappa\cdot\alpha\cdot r}$ as follows. The user computes

$$\mathsf{Aggregate}(\{g^{\frac{r}{\gamma+\tau(\mathsf{at}_i)}}, \tau(\mathsf{at}_i)\}_{\mathsf{at}_i\in S_1}) = g^{\frac{r}{\prod_{\mathsf{at}_i\in S_1}\gamma+\tau(\mathsf{at}_i)}}.$$

With the output the user computes $e(g,h)^{\kappa\cdot\alpha\cdot r} = e(g^{\frac{r}{\prod_{\mathsf{at}_i\in S_1}\gamma+\tau(\mathsf{at}_i)}}, E_0)$. After that, the user computes $e(g,h)^{\kappa\cdot\alpha} = e(C_1, h^{\frac{r-1}{\gamma}})\cdot e(g,h)^{\kappa\cdot\alpha\cdot r}$. Finally, the user recovers the message $M = \frac{C_M}{e(g,h)^{\kappa\cdot\alpha}}$.

5.2 Security Analysis

In this section, we prove that our scheme is secure against selective chosen-ciphertext attack, assuming that the (n,s)-$\mathsf{aMSE\text{-}DDH}_B$ problem is hard to solve.

Theorem 2. *Let λ be an integer. For any adversary $\mathcal{A}$ against the* $\mathsf{IND\text{-}sCPA}$ *security of our CP-ABE-AR encryption scheme $\mathcal{S}_{AR}$, for an attribute universe $\mathcal{P}$ of size n, and a challenge set$S$ with $s = |S|$, there exists an algorithm B of the* (n,s)-$\mathsf{aMSE\text{-}DDH}_B$ *problem, such that*

$$\mathsf{Adv}_{\mathcal{B}}^{(n,s)\text{-}\mathsf{aMSE\text{-}DDH}_B}(\lambda) \geq \mathsf{Adv}_{\mathcal{A},\mathcal{S}_{AR}}^{\mathsf{IND\text{-}sCPA}}(\lambda).$$

We now give the details of the simulation.

Init $\mathcal{B}$ defines an attribute universe $\mathcal{P} = \{\mathsf{at}_1, \ldots, \mathsf{at}_n\}$ of cardinal n. $\mathcal{A}$ gives $\mathcal{B}$ the challenge access structure $\mathbb{A}^*$ defined by an AND-gate policy $\bigwedge_{\mathsf{at}\in S}\mathsf{at}$ where $S \subset \mathcal{P}$ of respective cardinal s. Here we assume $S = \{\mathsf{at}_{n-s+1}, \ldots, \mathsf{at}_n\}$.

Setup The algorithm $\mathcal{B}$ defines $g := g_0^{f(\gamma)}$, $h := h_0$. $\mathcal{B}$ then can compute

– the values $u_i = g^{\alpha\gamma^i} = g_0^{\alpha\gamma^i\cdot f(\gamma)}$ with line (2.2) of its input values, since the exponent $\alpha\cdot\gamma^i\cdot f(\gamma)$ is a linear combination of $\{g_2(\gamma)\cdot\alpha, \ldots, g_2(\gamma)\cdot\alpha\cdot\gamma^{2n-s}\}$ and $\mathcal{B}$ knows the coefficients of the exponent polynomial.
– the value $v = e(g,h)^\alpha = e(g_0^{\alpha\cdot f(\gamma)}, h_0)$ with line (2.2) for $g_0^{\alpha\cdot f(\gamma)}$ and line (2.4) for h_0.
– elements in $\{h^{\alpha\gamma^i} = h_0^{\alpha\cdot\gamma^i}\}_{i=0,\ldots,n}$ with line (2.6).

– The encoding τ is defined as $\tau(\mathsf{at}_i) = x_i$ for $i = 1, \ldots, n$. It can be seen that the encodings of the first $n-s$ elements are the opposite of the roots of $f(X)$, the encodings of the attributes in S are the opposite of roots of $g(X)$.

Finally, $\mathcal{B}$ sends to $\mathcal{A}$ the simulated public parameters: $\Big(u, v, h, \{h^{\alpha\gamma^i}\}_{i=0,\ldots,n}, \tau\Big)$.

Phase 1 The adversary $\mathcal{A}$ makes private key queries. To respond to a query on attribute set $W \subset \mathcal{P}$, where $W \not\models \mathbb{A}^*$, the algorithm $\mathcal{B}$ must produce a tuple of the form $\Big(\{g^{\frac{r}{\gamma+\tau(\mathsf{at})}}\}_{\mathsf{at}\in W}, h^{\frac{r-1}{\gamma}}\Big)$.

Observe that since $W \not\models \mathbb{A}^*$ all allowed queries must satisfy $|W_S| < s$. $\mathcal{B}$ defines the polynomial $Q_{W_S}(X) = \begin{cases} 1 & |W_S| = 0 \\ \lambda_i \cdot \prod_{\mathsf{at} \in W_S}(X + \tau(\mathsf{at})) & |W_S| > 0 \end{cases}$, where $\lambda = \left(\prod_{A \in \omega_S} \tau(\mathsf{at})\right)^{-1}$, and simulates a private key for W as follows:

$\mathcal{B}$ picks at random y_W in $\mathbb{Z}_p$, and defines $r := (1 + \omega y_W \gamma) Q_{W_S}(\gamma)$. $\mathcal{B}$ then computes the elements for sk_W:

– For any attribute $\mathsf{at} \in W$, $g^{\frac{r}{\gamma+\tau(\mathsf{at})}} = g_0^{\omega\gamma y_W \cdot \frac{f(\gamma)Q_{W_S}(\gamma)}{\gamma+\tau(\mathsf{at})}} \cdot g_0^{\frac{f(\gamma)Q_{W_S}(\gamma)}{\gamma+\tau(\mathsf{at})}}$. Since an attribute $\mathsf{at} \in W$ can be in W_S or $\mathcal{P} \setminus (S)$, $(\gamma + \tau(\mathsf{at}))|f(\gamma)g_2(\gamma)Q_{W_S}(\gamma)$. The first factor can be computed with line (2.3) as its exponent is a polynomial in γ of degree at most $n-1$, and the second factor can be computed with line (2.1) as its exponent is a polynomial in γ of degree at most $n-2$.

– The value $h^{\frac{r-1}{\gamma}} = h_0^{\omega y_W Q_{W_S}(\gamma)} \cdot h_0^{\frac{Q_{W_S}(\gamma)-1}{\gamma}}$, where the first factor can be computed from line (2.7) and the second factor can be computed from line (2.4), since $Q_{W_{S_1}}(\gamma)$ is a polynomial with independent term 1 by its definition, thus $\frac{Q_{W_S}(\gamma)-1}{\gamma}$ is a linear combination of $\{1, \gamma, \ldots, \gamma^{s-2}\}$.

Challenge Once $\mathcal{A}$ sends to $\mathcal{B}$ the two messages M_0 and M_1 as well as a attribute set $\mathcal{U}^*$ with $t = |\mathcal{U}^*|$ and $\mathcal{U}^* \cap S = \emptyset$ including all attributes needed to be revoked, $\mathcal{B}$ flips a coin $\beta \in \{0, 1\}$, and sets $C_M^* = T_0 \cdot M_\beta$. To simulate the rest of the ciphertext components, $\mathcal{B}$ sets $E_0^* = h_0^{\kappa \cdot g(\gamma)}$ which is given in line (2.5). Then, B computes $C_1^* = (g_0^{\kappa\gamma f(\gamma)})^{-1}$ from line (2.1). $\mathcal{B}$ gives $\mathcal{A}$ the challenge ciphertext $CT^* = (E_0^*, C_1^*, C_M^*)$.

Here we observe that

if $\mathcal{U}^* = \emptyset$**,** $t = 0$ $\mathcal{B}$ should output to the adversary $CT = \mathsf{Enc}(\mathsf{params}, \mathbb{A}^*, 0, M_\beta) = (E_0, C_1, C_M)$ for access structure $\mathbb{A}^*$, of which the challenge ciphertext matches the form;

if $\mathcal{U}^* \neq \emptyset$ $\mathcal{B}$ should output $CT' = \mathsf{Revoke}(\mathsf{params}, \mathsf{Enc}(\mathsf{params}, \mathbb{A}', t, M_\beta), \mathcal{U}^*) = (E_0', C_1', C_M)$ for access structure $\mathbb{A}^*$, of which the challenge ciphertext matches the form as well.

Phase 2 After the challenge step $\mathcal{A}$ may make other key extraction queries, which are answered as before.

Guess $\mathcal{A}$ outputs a β'. If $\beta' = \beta$, $\mathcal{B}$ outputs 0; otherwise $\mathcal{B}$ outputs 1.

Probability Analysis:

Let $\mathcal{I} = (\overrightarrow{x}_n, \gamma, \kappa, \omega, \alpha, T_b, T_{1-b})$ be the input of the algorithm $\mathcal{B}$ and the adversary $\mathcal{A}$ break our CP-ABE scheme with advantage $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{IND\text{-}sCPA}}(\lambda)$. Below we analyse the simulation in two cases.

Case 1 ($\mathcal{U}^* = \emptyset$). Let $\kappa^* = \kappa \cdot \alpha$. One can verify that in this case, $E_0^* = h_0^{\kappa \cdot g(\gamma)} = h^{\kappa^* \cdot \alpha \cdot \gamma \cdot \prod_{\mathsf{at} \in S}(\gamma + \tau(\mathsf{at}))}$ and $C_1^* = g_0^{-\kappa \cdot \gamma \cdot f(\gamma)} = g_0^{-\kappa^* \cdot \alpha \cdot \gamma \cdot f(\gamma)} = u_1^{-\kappa^*}$. As for the C_M^*, we also note that if $b = 0$, $T_0 = e(g_0, h_0)^{\kappa f(\gamma)}$, then $C_M^* = e(g_0, h_0)^{\kappa f(\gamma)} \cdot M_\beta =$

$e(g^{\alpha}, h)^{\kappa^*} \cdot M_{\beta} = v^{\kappa^*} \cdot M_{\beta}$. Therefore, the simulation of $\mathcal{B}$ is perfect, and the adversary $\mathcal{A}$ will guess the bit β with its advantage. Hence, if $b = 0$ we have

$$|\Pr[\mathcal{B}(\mathcal{I}) = 0 | b = 0] - \frac{1}{2}| = \mathsf{Adv}_{\mathcal{A}}^{\mathsf{IND\text{-}sCPA}}(\lambda).$$

Else, if $b = 1$ and T_0 is uniformly random in $\mathbb{G}_T$, C_M^* is uniformly random and independent in $\mathbb{G}_T$, and the value of β is independent from $\mathcal{A}$'s view as well,

$$\Pr[\mathcal{B}(\mathcal{I}) = 0 | b = 1] = \frac{1}{2}.$$

Thus, we have the advantage of $\mathcal{B}$ in solving the (n, s)-aMSE-DDH$_B$ problem in Case 1 is

$$\begin{aligned}\mathsf{Adv}_{\mathcal{B}}^{\mathsf{(n,s)-aMSE_B\text{-}DDH}}(\lambda) &= |\Pr[\mathcal{B}(\mathcal{I}) = 0 | b = 0] - \Pr[\mathcal{B}(\mathcal{I}) = 0 | b = 1]| \\ &\geq \mathsf{Adv}_{\mathcal{A}}^{\mathsf{IND\text{-}sCPAwAR}}(\lambda).\end{aligned}$$

Case 2 ($\mathcal{U}^* \neq \emptyset$). In this case, we first show how a challenge ciphertext should be produced in a real game. Formally, the correct procedures are as follows.

Let $S' = \mathcal{U}^* \cup S$. The encryption algorithm $\mathsf{Enc}(\mathsf{params}, \mathbb{A}' = \bigwedge_{\mathsf{at} \in S'} \mathsf{at}, t, M_{\beta})$ is run to get CT^*. More precisely, it picks a randomness $\kappa' \in \mathbb{Z}_p$ and computes,

$$\begin{aligned}CT^* &= (E_0^*, C_1^*, \cdots, C_{t+1}^*, C_M^*) \\ &= (h^{\kappa' \cdot \alpha \cdot \prod_{\mathsf{at} \in S_1} (\gamma + \tau(\mathsf{at}))}, u_1^{-\kappa'}, \ldots, u_{t+1}^{-\kappa'}, C_M = v^{\kappa'} \cdot M).\end{aligned}$$

The revocation algorithm $\mathsf{Revoke}(\mathsf{params}, CT^*, \mathcal{U}^*)$ is run to revoke the attribute set $\mathcal{U}^*$ from the access policy of the ciphertext CT^*. It processes as follows.

Let $F(x)$ be the polynomial in x as

$$F(x) = \frac{1}{\prod_{\mathsf{at}' \in \mathcal{U}^*} \tau(\mathsf{at}')} \prod_{\mathsf{at}' \in \mathcal{U}^*} (x + \tau(\mathsf{at}')) = f_t x^t + f_{t-1} x^{t-1} + \cdots + f_0.$$

Compute $C_M' = C_M \cdot e(\prod_{i=1}^{l} C_i^{f_i}, h) = M_{\beta} \cdot v^{\kappa' \cdot F(\gamma)}$.

Compute $E_0' = E_0^{\frac{1}{\prod_{\mathsf{at}' \in \mathcal{U}} \tau(\mathsf{at}')}} = h^{\kappa' \cdot \alpha \cdot F(\gamma) \cdot \prod_{\mathsf{at} \in S} (\gamma + \tau(\mathsf{at}))}$.

Compute $C_1' = \prod_{i=1}^{l+1} C_i^{f_{i-1}} = u_1^{-\kappa' \cdot F(\gamma)}$.

Finally, the challenge ciphertext in a real game is produced $CT' = (E_0', C_1', C_M')$.

Now we assume that the randomness κ' used in producing CT^* is defined as $\kappa' = \frac{\kappa}{\alpha} \cdot \frac{1}{F(\gamma)}$. Then let $\kappa^* = \kappa / \alpha$ and the challenge ciphertext CT' turns out to be as follows,

$$C_M' = M_{\beta} \cdot v^{\frac{\kappa}{\alpha}} = M_{\beta} \cdot v^{\kappa^*},$$

$$E_0' = h^{\kappa \cdot \prod_{\mathsf{at} \in S} (\gamma + \tau(\mathsf{at}))} = h^{\kappa^* \cdot \alpha \cdot \gamma \cdot \prod_{\mathsf{at} \in S} (\gamma + \tau(\mathsf{at}))},$$

$$C_1' = u_1^{\frac{-\kappa}{\alpha}} = u_1^{\kappa^*}.$$

It can be seen that if $b = 0$, $T_0 = e(g_0, h_0)^{\kappa \cdot f(\gamma)}$, the challenge ciphertext in a real game is exactly the same as the simulated challenge ciphertext. The simulated game

would be a perfect simulation if it can be proved that the setting of κ' is indistinguishable from a real random value from the view of $\mathcal{A}$. It will suffice as κ is random to $\mathcal{A}$. Thus, if $b = 0$ we have

$$|\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 0] - \frac{1}{2}| = \mathsf{Adv}^{\mathsf{IND\text{-}sCPA}}_{\mathcal{A},\mathcal{S}_{AR}}(\lambda).$$

On the other hand, if $b = 1$ and T_0 is a random element from $\mathbb{G}_T$, C^*_M is random and independent from the view of $\mathcal{A}$, $\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 1] = \frac{1}{2}$. Thus, we have the advantage of $\mathcal{B}$ in solving the (n, s)-aMSE-DDH$_B$ problem in Case 2 is

$$\begin{aligned}\mathsf{Adv}^{\mathsf{(n,s)-aMSE_B\text{-}DDH}}_{\mathcal{B}}(\lambda) &= |\Pr[\mathcal{B}(\mathcal{I}) = 0|b = 0] - \Pr[\mathcal{B}(\mathcal{I}) = 0|b = 1]| \\ &\geq \mathsf{Adv}^{\mathsf{IND\text{-}sCPAwAR}}_{\mathcal{A},\mathcal{S}_{AR}}(\lambda).\end{aligned}$$

This completes the proof. □.

6 Intractability of (n, s)-aMSE-DDH Assumptions

In this section, we provide the analysis of the intractability of (n, s)-aMSE-DDH problem. The intractability analysis is based on the analysis in the generic group model in [7].

6.1 Notations

For simplicity, we scope the problem to bilinear map group systems in the symmetric case ($\mathbb{G}_1 = \mathbb{G}_2 = \mathbb{G}$). Let then $\mathbb{S} = (p, \mathbb{G}, \mathbb{G}, \mathbb{G}_T, e(\cdot, \cdot))$ be a bilinear map group system. Let $g \in \mathbb{G}$ be a generator of $\mathbb{G}$, and set $g_T = e(g, g) \in \mathbb{G}_T$. Let s, m be two positive integers and $P, Q \in \mathbb{F}_p[X_1, \ldots, X_m]^s$ be two lists containing s m-variate polynomials over $\mathbb{F}_p$. Thus, P and Q can be written as $P = (p_1, p_2, \ldots, p_s)$ and $Q = (q_1, q_2, \ldots, q_s)$, and impose that $p_1 = q_1 = 1$. For any function $h : \mathbb{F}_p \to \Omega$ and vector $(x_1, ..., x_m) \in \mathbb{F}^m_p$, the notation $h(P(x_1, \ldots, x_m))$ stands for $(h(p_1(x_1, \ldots, x_m)), \ldots, h(p_s(x_1, \ldots, x_m))) \in \Omega^s$.

We use a similar notation for the s-tuple Q. Let $f \in \mathbb{F}_p[X_1, ..., X_m]$. It is said that f depends on (P, Q), which we denote by $f \in \langle P, Q\rangle$, when there exists a linear decomposition $f = \sum_{1\leq i,j\leq s} a_{i,j} \cdot p_i \cdot p_j + \sum_{1\leq i\leq s} b_i \cdot q_i$, where $a_{i,j}, b_i \in \mathbb{Z}_p$. Let P, Q be as above and $f \in \mathbb{F}_p[X_1, \ldots, X_m]$. The (P, Q, f)-General Diffie-Hellman Exponent problems are defined as follows.

Definition 2 ((P, Q, f)-GDHE[4]). *Given the tuple*

$$H(x_1, ..., x_m) = \left(g^{P(x_1,...,x_m)}, g_T^{Q(x_1,...,x_m)}\right) \in \mathbb{G}^s \times \mathbb{G}_T^s,$$

compute $g^{f(x_1,...,x_m)}$.

Definition 3 ((P, Q, f)-GDDHE). *Given* $H(x_1, ..., x_m) \in \mathbb{G}^s \times \mathbb{G}_T^s$ *as above, and* $T \in \mathbb{G}_T$*, decide whether* $T = g^{f(x_1,...,x_m)}$.

We refer to [4] for a proof that (P, Q, f)-GDHE and (P, Q, f)-GDDHE have generic security when $f \notin \langle P, Q\rangle$. We will prove that our construction is secure by first exhibiting the polynomials P, Q and f involved in the security proofs, and then by showing that $f \notin \langle P, Q\rangle$.

6.2 (n, s)-aMSE-DDH

In this section, we prove the intractability of distinguishing the two distributions involved in the (n, s)-aMSE-DDH$_A$ problem (cf. Corollary 1, Section 3.2). The proof of the intractability of the (n, s)-aMSE-DDH$_B$ problem (cf. Corollary 2, Section 3.2) is similar to that for Corollary 1, and hence, we omit it.

Proof (Proof of Corollary 1). To wrap up Corollary 1, we need to show that (n, s)-aMSE-DDH$_A$ problem fits in the framework of Theorem A.2 in [4]. As mentioned above, we consider our problem in the weakest case $\mathbb{G}_1 = \mathbb{G}_2 = \mathbb{G}$ and pose $g_0 = g, h_0 = g^\beta$. Our problem can be reformulated as (P, Q, F)-GDDHE where

$$
\begin{aligned}
P &= \begin{pmatrix}
1, \gamma, \ldots, \gamma^{n-2}, \\
\kappa \cdot \gamma \cdot f(\gamma), \\
\alpha, \alpha \cdot \gamma, \ldots, \alpha \cdot \gamma^{n-s+1}, \\
\omega' \cdot \gamma, \omega \cdot \gamma^2, \ldots, \omega\gamma^{n-1}, \\
\beta, \beta \cdot \gamma, \ldots, \beta \cdot \gamma^{s-2}, \\
\beta\kappa \cdot g(\gamma), \beta\kappa \cdot \gamma \cdot g(\gamma), \ldots, \beta\kappa \cdot \gamma^{n-s} \cdot g(\gamma), \\
\beta\alpha, \beta\alpha \cdot \gamma, \ldots, \beta \cdot \alpha \cdot \gamma^n \\
\beta\omega, \beta\omega \cdot \gamma, \ldots, \beta\omega \cdot \gamma^n,
\end{pmatrix} \\
Q &= (1) \\
F &= \beta\kappa \cdot f(\gamma).
\end{aligned}
$$

We need to prove the independence of F from $\langle P, Q\rangle$. By making all possible products of two polynomials from P which are multiples of $\beta\kappa$, we want to prove that the sum of any polynomials from the list R below does not lead to F:

$$
R = \begin{cases}
\beta\kappa \cdot \gamma \cdot A(\gamma) f(\gamma) \\
\beta\kappa \cdot B(\gamma) g(\gamma) \\
\beta\kappa \cdot \gamma \cdot B(\gamma) g(\gamma) \\
\vdots \\
\beta\kappa \cdot \gamma^{s-2} \cdot B(\gamma) g(\gamma)
\end{cases}
$$

where A, B are polynomials in γ.

After simplifying the list R, it can be seen that if F is not independent of $\langle P, Q\rangle$ we can then derive $\gamma \cdot f(\gamma)$ from following list: $R' = \begin{cases} \gamma \cdot A(\gamma) f(\gamma) \\ B'(\gamma) g(\gamma) \end{cases}$ where A, B' are polynomials in γ with $0 \leq \deg A \leq s-2, 0 \leq \deg B' \leq n+s-4$.

Thus, we have the following equation:

$$
f(\gamma) = \gamma \cdot A(\gamma) f(\gamma) + B'(\gamma) g(\gamma)
$$

which can then be re-written into $(1 - \gamma \cdot A(\gamma)) f(\gamma) = B'(\gamma) g(\gamma)$ where $1 - \gamma \cdot A(\gamma) \neq 0$, $\deg B_1(\gamma) \leq n + s - 4$. Since f and g are coprime, we must have $g(\gamma) | (1 - \gamma \cdot A(\gamma))$. However, $\deg(1 - \gamma \cdot A(\gamma)) < \deg g(\gamma)$ will result in $1 - \gamma \cdot A(\gamma) = 0$, which contradicts with the fact $1 - \gamma \cdot A(\gamma) \neq 0$. □

7 Conclusion

In this paper, we considered the problem of access policy update in ABE schemes, which make the ABE schemes become practical. When an ABE scheme is not equipped with efficient access policy update, it cannot be used in practice as policy update is an essential feature in the dynamic environment. We outlined some trivial solutions including using AB-PRE system, and also pointed out the difference between access policy update and ciphertext re-encryption, which showed the importance of a general efficient access policy update mechanism. We presented notions of ciphertext-policy attribute-based encryption supporting attribute addition and revocation, and subsequently presented two new CP-ABE schemes featured with functionalities of adding and revoking attributes, respectively. We also proposed a new selective CPA model for CP-ABE with these new features. Finally, we also proved the security of our schemes. The proposed schemes are proven secure against selective CPA under the assumptions that the augmented Multi-Sequence of Exponents Decisional Diffie-Hellman (aMSE-DDH) problems are hard. The intractability of the aMSE-DDH problems is proved in generic group model within the frame work of General Diffie-Hellman Exponent problem in [4]. It remains an open problem to obtain a scheme integrated with efficient access policy update mechanism supporting more expressive access policies which can be proven secure under a more general computational assumption.

Acknowledgement

This work is partially supported by ARC Project (DP130101383).

References

1. Attrapadung, N., Herranz, J., Laguillaumie, F., Libert, B., De Panafieu, E., Ràfols, C.: Attribute-based encryption schemes with constant-size ciphertexts. Theoretical Computer Science 422, 15–38 (2012)
2. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: Security and Privacy, 2007. SP'07. IEEE Symposium on. pp. 321–334. IEEE (2007)
3. Blaze, M., Bleumer, G., Strauss, M.: Divertible protocols and atomic proxy cryptography. In: Advances in CryptologyEUROCRYPT'98, pp. 127–144. Springer (1998)
4. Boneh, D., Boyen, X., Goh, E.J.: Hierarchical identity based encryption with constant size ciphertext. In: Advances in Cryptology–EUROCRYPT 2005, pp. 440–456. Springer (2005)
5. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Advances in CryptologyCRYPTO 2001. pp. 213–229. Springer (2001)
6. Cheung, L., Newport, C.: Provably secure ciphertext policy abe. In: Proceedings of the 14th ACM conference on Computer and communications security. pp. 456–465. ACM (2007)
7. Delerablée, C., Pointcheval, D.: Dynamic threshold public-key encryption. In: Advances in Cryptology–CRYPTO 2008, pp. 317–334. Springer (2008)
8. Goyal, V., Jain, A., Pandey, O., Sahai, A.: Bounded ciphertext policy attribute based encryption. In: Automata, languages and programming, pp. 579–591. Springer (2008)
9. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM conference on Computer and communications security. pp. 89–98. Acm (2006)

10. Herranz, J., Laguillaumie, F., Ràfols, C.: Constant size ciphertexts in threshold attribute-based encryption. In: Public Key Cryptography–PKC 2010, pp. 19–34. Springer (2010)
11. Lewko, A., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In: Advances in Cryptology–EUROCRYPT 2010, pp. 62–91. Springer (2010)
12. Lewko, A., Waters, B.: New proof methods for attribute-based encryption: Achieving full security through selective techniques. In: Advances in Cryptology–CRYPTO 2012, pp. 180–198. Springer (2012)
13. Li, J., Ren, K., Zhu, B., Wan, Z.: Privacy-aware attribute-based encryption with user accountability. In: Information Security, pp. 347–362. Springer (2009)
14. Liang, K., Au, M.H., Liu, J.K., Susilo, W., Wong, D.S., Yang, G., Yu, Y., Yang, A.: A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. Future Generation Computer Systems 52, 95–108 (2015)
15. Liang, K., Au, M.H., Susilo, W., Wong, D.S., Yang, G., Yu, Y.: An adaptively cca-secure ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. In: Information Security Practice and Experience, pp. 448–461. Springer (2014)
16. Liang, K., Fang, L., Susilo, W., Wong, D.: A ciphertext-policy attribute-based proxy re-encryption with chosen-ciphertext security. In: Intelligent Networking and Collaborative Systems (INCoS), 2013 5th International Conference on. pp. 552–559. IEEE (2013)
17. Liang, X., Cao, Z., Lin, H., Shao, J.: Attribute based proxy re-encryption with delegating capabilities. In: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security. pp. 276–286. ACM (2009)
18. Luo, S., Hu, J., Chen, Z.: Ciphertext policy attribute-based proxy re-encryption. In: Information and Communications Security, pp. 401–415. Springer (2010)
19. Naor, M.: On cryptographic assumptions and challenges. In: Advances in Cryptology-CRYPTO 2003, pp. 96–109. Springer (2003)
20. Nishide, T., Yoneyama, K., Ohta, K.: Attribute-based encryption with partially hidden encryptor-specified access structures. In: Applied cryptography and network security. pp. 111–129. Springer (2008)
21. Ostrovsky, R., Sahai, A., Waters, B.: Attribute-based encryption with non-monotonic access structures. In: Proceedings of the 14th ACM conference on Computer and communications security. pp. 195–203. ACM (2007)
22. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Advances in Cryptology–EUROCRYPT 2005, pp. 457–473. Springer (2005)
23. Seo, H.J., Kim, H.W.: Attribute-based proxy re-encryption with a constant number of pairing operations. Journal of information and communication convergence engineering 10(1), 53–60 (2012)
24. Susilo, W., Chen, R., Guo, F., Yang, G., Mu, Y., Chow, Y.W.: Recipient revocable identity-based broadcast encryption. In: ASIACCS (2016)
25. Waters, B.: Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In: Public Key Cryptography–PKC 2011, pp. 53–70. Springer (2011)
26. Waters, B.: Functional encryption for regular languages. In: Advances in Cryptology–CRYPTO 2012, pp. 218–235. Springer (2012)
27. Zhang, Y., Chen, X., Li, J., Wong, D.S., Li, H.: Anonymous attribute-based encryption supporting efficient decryption test. In: Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security. pp. 511–516. ACM (2013)