Fully Collusion-Resistant Traceable Key-Policy Attribute-Based Encryption with Sub-linear Size Ciphertexts

Zhen Liu[1], Zhenfu Cao[2], and Duncan S. Wong[1]

[1] City University of Hong Kong, Hong Kong SAR, China.
zhenliu7@cityu.edu.hk, duncan@cityu.edu.hk
[2] Shanghai Jiao Tong University, Shanghai, China.
zfcao@cs.sjtu.edu.cn

Abstract. Recently a series of expressive, secure and efficient Attribute-Based Encryption (ABE) schemes, both in key-policy flavor and ciphertext-policy flavor, have been proposed. However, before being applied into practice, these systems have to attain traceability of malicious users. As the decryption privilege of a decryption key in Key-Policy ABE (resp. Ciphertext-Policy ABE) may be shared by multiple users who own the same access policy (resp. attribute set), malicious users might tempt to leak their decryption privileges to third parties, for financial gain as an example, if there is no tracing mechanism for tracking them down. In this work we study the traceability notion in the setting of Key-Policy ABE, and formalize Key-Policy ABE supporting fully collusion-resistant blackbox traceability. An adversary is allowed to access an arbitrary number of keys of its own choice when building a decryption-device, and given such a decryption-device while the underlying decryption algorithm or key may not be given, a Blackbox tracing algorithm can find out at least one of the malicious users whose keys have been used for building the decryption-device. We propose a construction, which supports both fully collusion-resistant blackbox traceability and high expressiveness (i.e. supporting any monotonic access structures). The construction is fully secure in the standard model (i.e. it achieves the best security level that the conventional non-traceable ABE systems do to date), and is efficient that the fully collusion-resistant blackbox traceability is attained at the price of making ciphertexts grow only sub-linearly in the number of users in the system, which is the most efficient level to date.

Keywords: Attribute-Based Encryption, Key-Policy, Blackbox Traceability, Efficiency

1 Introduction

Attribute-based encryption (ABE), as a promising tool for fine-grained access control on encrypted data, has attracted much attention since its introduction by Sahai and Waters [24] in 2005, and work has been done to achieve better expressivity, security and efficiency, in both key-policy flavor [10,22,14,21,1,27,7,23] and ciphertext-policy flavor [6,9,26,15,21,11,16,23]. Due to their high expressivity of access policy and efficient one-to-many encryption, both Key-Policy ABE (KP-ABE) and Ciphertext-Policy ABE (CP-ABE) have extensive applications. For example, in a KP-ABE system (CP-ABE proceeds the other way around), each private key is associated with an access policy over descriptive attributes issued by an authority, each ciphertext is associated with an attribute set specified by the encryptor, and if and only if the attribute set of a ciphertext satisfies the access policy, the private key can decrypt the ciphertext. In a pay-TV system the television broadcaster can encrypt the broadcast using the descriptive attributes, such as the name of the program ("The Big Bang Theory"), the genre ("drama"), the season, the episode number, the year, the month, etc. And a subscriber may determine and pay for his subscribing policy, such as "(The Big Bang Theory OR Criminal Minds) AND 2014".

Recently, the expressivity, security and efficiency of ABE have been relatively well developed. The KP-ABE systems in [14,21] and the CP-ABE systems in [15,21,16][3] are highly expressive (i.e. supporting any monotonic access structures), fully secure in the standard model, and satisfactorily efficient. However, to apply these systems into practice, the traceability of malicious users is needed. In an ABE system in general, as a decryption privilege could be possessed by multiple users who own the same access policy (in KP-ABE) or attribute set (in CP-ABE), malicious users might tempt to leak their decryption privileges to third parties, for financial gain as an example, if there is no tracing mechanism for finding these malicious users out. For example, both user Alex with access policy "(The Big Bang Theory OR Criminal Minds) AND 2014" and user Bob with access policy "(The Big Bang Theory OR CSI) AND 2014" might be the malicious user who builds and sells a decryption blackbox that can decrypt the ciphertexts generated under attributes {The Big Bang Theory, 2014}.

While all the aforementioned ABE systems suffer from this problem, some recent attempts [18,28,17,25,13,20,19] have been made to achieve traceability. Specifically, there are two levels of traceability: (1) given a well-formed decryption key, a *Whitebox* tracing algorithm can find out the original key owner; and (2) given a decryption-device while the underlying decryption algorithm or key may not be given, a *Blackbox* tracing algorithm, which treats the decryption-device as an oracle, can find out at least one of the malicious users whose keys have been used for building the decryption-device. Furthermore, a system is said to support *fully collusion-resistant blackbox traceability* if an adversary can access an arbitrary number of keys (in other words, when an arbitrary number of malicious users collude) when building the decryption-device, and is said to support *t-collusion-resistant blackbox traceability*, if an adversary is restricted from getting more than t decryption keys when building the decryption-device. While the Blackbox Traceable CP-ABE scheme in [19] is highly expressive, fully secure in the standard model, and efficient in achieving fully collusion-resistant blackbox traceability at the expense of overhead sub-linear in the number of users, there is no traceable KP-ABE scheme achieving comparable expressivity, security and efficiency. In particular, (1) the blackbox traceability of [28] is 1-collusion-resistant (i.e. cannot resist collusion attack); (2) [25] only supports single threshold policy and t-collusion-resistant blackbox traceability; (3) the fully collusion-resistant blackbox traceable predicate encryption scheme in [13] implies an expressive KP-ABE scheme (at the cost of converting monotonic access structure into DNF, which will result in larger ciphertext size), but the overhead for traceability is linear in the number of users; and (4) all the schemes in [28,25,13] are only selectively secure.

	Adaptively Secure	Highly Expressive	Fully Collusion-Resistant Traceable	Overhead for Traceability
[28]	×	√	×	$O(\log \mathcal{K})$
[25]	×	×	×	$O(t^2 \log \mathcal{K} + \log(1/\epsilon))$ [1]
[13]	×	√	√	$O(\mathcal{K})$
this paper	√	√	√	$O(\sqrt{\mathcal{K}})$

[1] [25] is only t-collusion-resistant traceable and the large overhead $O(t^2 \log \mathcal{K} + \log(1/\epsilon))$ makes the scheme impractical. Furthermore, to achieve fully collusion-resistant traceability (i.e., $t = \mathcal{K}$, the number of users in the system), the overhead of the scheme will be $O(\mathcal{K}^2 \log \mathcal{K} + \log(1/\epsilon))$. ϵ is the probability of error that a colluder is not traced.

Table 1. Comparison with existing Traceable KP-ABE schemes

[3] [14] is the full version of [15], where [15] proposed expressive, fully secure and efficient CP-ABE schemes, [14] further proposed an expressive, fully secure and efficient KP-ABE scheme additionally.

1.1 Our Results

In this paper, we first formalize the fully collusion-resistant blackbox traceability notions for expressive KP-ABE, then we formalize a simpler primitive called Augmented KP-ABE and show that a secure Augmented KP-ABE can be directly transformed into a Traceable KP-ABE. With such a transformation, to obtain a fully collusion-resistant blackbox traceable KP-ABE scheme, we propose an Augmented KP-ABE construction, implying a traceable KP-ABE construction that is fully secure in the standard model, highly expressive in supporting any monotonic access structures, and efficient in achieving fully collusion-resistant blackbox traceability at the expense of having the ciphertext size be sub-linear in the number of users in the system. In Table 1 we compare our traceable KP-ABE scheme with existing traceable KP-ABE schemes in literature.

Paper Organization. In Sec. 2 we formalize the fully collusion-resistant blackbox traceability notions for expressive KP-ABE. Then in Sec. 3, we propose a primitive called Augmented KP-ABE, and show that an Augmented KP-ABE with message-hiding and index-hiding properties implies a secure KP-ABE with traceability. Finally in Sec. 4, we propose a concrete construction of Augmented KP-ABE and show that it is message-hiding and index-hiding.

2 KP-ABE with Traceability

We first review the definition of KP-ABE which is based on conventional (non-traceable) KP-ABE (e.g. [10,14]) with the exception that in our 'functional' definition, we explicitly assign and identify users using unique indices, and let $\mathcal{K}$ be the number of users in a KP-ABE system. Then we introduce the fully collusion-resistant traceability definition against attributes-specific decryption blackbox, which reflects most practical applications.

2.1 KP-ABE

Before defining KP-ABE system, we first provide some background about access policy in the context of KP-ABE.

Definition 1. (Access Structure) *[2] Let $\mathcal{U} = \{a_1, a_2, \ldots, a_n\}$ be a set of attributes. A collection $\mathbb{A} \subseteq 2^{\mathcal{U}}$ is monotone if $\forall B, C : B \in \mathbb{A}$ and $B \subseteq C$ imply $C \in \mathbb{A}$. An access structure (resp., monotone access structure) is a collection (resp., monotone collection) $\mathbb{A}$ of non-empty subsets of $\mathcal{U}$, i.e., $\mathbb{A} \subseteq 2^{\mathcal{U}} \setminus \{\emptyset\}$. The sets in $\mathbb{A}$ are called authorized sets, and the sets not in $\mathbb{A}$ are called unauthorized sets. Also, for an attribute set $S \subseteq U$, if $S \in \mathbb{A}$ then we say S satisfies the access structure $\mathbb{A}$, otherwise we say S does not satisfy $\mathbb{A}$.*

Unless stated otherwise, by an access structure we mean a monotone access structure for the rest of this paper.

For simplicity, for a positive integer, for example n, we use the notation $[n]$ to denote the set $\{1, 2, \ldots, n\}$. A Key-Policy ABE (KP-ABE) scheme consists of the following four algorithms:

- $\mathsf{Setup}(\lambda, \mathcal{U}, \mathcal{K}) \rightarrow (\mathsf{PP}, \mathsf{MSK})$. The algorithm takes as input a security parameter $\lambda \in \mathbb{N}$, the attribute universe (i.e., the set of attributes) $\mathcal{U}$, and the number of users $\mathcal{K}$ in the system, it outputs a public parameter PP and a master secret key MSK.
- $\mathsf{KeyGen}(\mathsf{PP}, \mathsf{MSK}, \mathbb{A}) \rightarrow \mathsf{SK}_{k,\mathbb{A}}$. The algorithm takes as input PP, MSK, and an access structure $\mathbb{A}$, and outputs a private key $\mathsf{SK}_{k,\mathbb{A}}$, which is assigned and identified by a unique index $k \in [\mathcal{K}]$.

$\mathsf{Encrypt}(\mathsf{PP}, M, S) \rightarrow CT_S$. The algorithm takes as input PP, a message M, and an attribute set $S \subseteq \mathcal{U}$, and outputs a ciphertext CT_S such that only users whose access structures are satisfied by S can decrypt CT_S and recover M. S is implicitly included in CT_S.

$\mathsf{Decrypt}(\mathsf{PP}, CT_S, \mathsf{SK}_{k,\mathbb{A}}) \rightarrow M$ or $\perp$. The algorithm takes as input PP, a ciphertext CT_S associated with an attribute set S, and a private key $\mathsf{SK}_{k,\mathbb{A}}$. If S satisfies $\mathbb{A}$, the algorithm outputs message M, otherwise it outputs $\perp$ indicating the failure of decryption.

The security of the above KP-ABE scheme is defined using the following message-hiding game, which is a typical semantic security game and is based on that for conventional KP-ABE [10,14] security against adaptive adversaries, except that each key is explicitly identified by a unique index.

$\mathsf{Game}_{\mathsf{MH}}$. The **message-hiding game** is defined between a challenger and an adversary $\mathcal{A}$ as follows:

Setup. The challenger runs $\mathsf{Setup}(\lambda, \mathcal{U}, \mathcal{K})$ and gives the public parameter PP to $\mathcal{A}$.

Phase 1. For $i = 1$ to Q_1, $\mathcal{A}$ adaptively submits (index, access structure) pair $(k_i, \mathbb{A}_{k_i})$ to the challenger. The challenger responds with $\mathsf{SK}_{k_i,\mathbb{A}_{k_i}}$.

Challenge. $\mathcal{A}$ submits two equal-length messages M_0, M_1 and an attribute set S^*. The challenger flips a random coin $b \in \{0,1\}$, and and sends $CT_{S^*} \leftarrow \mathsf{Encrypt}(\mathsf{PP}, M_b, S^*)$ to $\mathcal{A}$.

Phase 2. For $i = Q_1 + 1$ to Q, $\mathcal{A}$ adaptively submits (index, access structure) pair $(k_i, \mathbb{A}_{k_i})$ to the challenger. The challenger responds with $\mathsf{SK}_{k_i,\mathbb{A}_{k_i}}$.

Guess. $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$ for b.

$\mathcal{A}$ wins the game if $b' = b$ under the **restriction** that S^* does not satisfy any of the queried access structures $\mathbb{A}_{k_1}, \ldots, \mathbb{A}_{k_Q}$. The advantage of $\mathcal{A}$ is defined as $\mathsf{MHAdv}_{\mathcal{A}} = |\Pr[b' = b] - \frac{1}{2}|$.

Definition 2. *A $\mathcal{K}$-user KP-ABE scheme is secure if for all probabilistic polynomial time (PPT) adversaries $\mathcal{A}$ the advantage $\mathsf{MHAdv}_{\mathcal{A}}$ is a negligible function of λ.*

It is worth noticing that: (1) although the index of each user private key is assigned by the KeyGen algorithm, to capture the security that an attacker can adaptively choose keys to corrupt, the above security model allows the adversary to specify the index when he makes a key query, i.e., for $i = 1$ to Q, the adversary submits (index, access structure) pair $(k_i, \mathbb{A}_{k_i})$ to query a private key for access structure $\mathbb{A}_{k_i}$, and the challenger will assign k_i to be the index of the private key, where $Q \leq \mathcal{K}$, $k_i \in [\mathcal{K}]$, and $k_i \neq k_j$ $\forall 1 \leq i \neq j \leq Q$ (this is to guarantee that each user/key can be *uniquely* identified by an index); and (2) for $k_i \neq k_j$ we do not require $\mathbb{A}_{k_i} \neq \mathbb{A}_{k_j}$, i.e., different users/keys may have the same access strcuture. We remark that these two points apply to the rest of the paper.

Remark: Compared with a conventional (non-traceable) KP-ABE [10,14], the above definition has the same Encrypt and Decrypt functionality, and almost the same Setup and KeyGen with only slight differences: predefining the number of users $\mathcal{K}$ in Setup and assigning each user a unique index $k \in [\mathcal{K}]$. Presetting the number of users is indeed a tradeoff but is also a necessary cost for achieving blackbox traceability. We stress that in practice, this should not incur much concern, and all the existing blackbox traceable systems (e.g. [4,5,8,25,13,19]) have the same setting. Also being consistent with the conventional definition of KP-ABE, the user indices are not used in normal encryption (i.e. the encryptors do not need to know the indices of any users in order to encrypt) and different users (with different indices) may have the same access policy. In summary, a secure KP-ABE system defined as above has all the appealing properties that a conventional KP-ABE system [10,14] has, that is, fully collusion-resistant security, fine-grained access control on encrypted data, and efficient one-to-many encryption. The unique index of each user/private key is to uniquely identify the users and allow the traceability.

2.2 KP-ABE Traceability

An attributes-specific decryption blackbox $\mathcal{D}$ in the setting of KP-ABE is viewed as a probabilistic circuit that can decrypt ciphertexts generated under some specific attribute set. *In particular, an attributes-specific decryption blackbox $\mathcal{D}$ is described with an attribute set $S_{\mathcal{D}}$ and a non-negligible probability value ϵ (i.e. $\epsilon = 1/f(\lambda)$ for some polynomial f), and this blackbox $\mathcal{D}$ can decrypt the ciphertexts generated under $S_{\mathcal{D}}$ with probability at least ϵ.* Such an attributes-specific decryption blackbox reflects most practical scenarios. In particular, once a decryption blackbox is found being able to decrypt some ciphertext with non-negligible probability (regardless of how this is found, for example, an explicit description of the blackbox's decryption ability is given, or the law enforcement agency finds some clue), we can regard it as an attributes-specific decryption blackbox with the corresponding attribute set (which is associated to the ciphertext).[4] And for a decryption blackbox, if multiple attribute sets are found that corresponding ciphertexts can be decrypted by this blackbox with non-negligible probability, we can regard the blackbox as multiple attributes-specific decryption blackbox, each with a different attribute set.

We now define a tracing algorithm against an attributes-specific decryption blackbox as follows.

$\mathsf{Trace}^{\mathcal{D}}(\mathsf{PP}, S_{\mathcal{D}}, \epsilon) \rightarrow \mathbb{K}_T \subseteq [\mathcal{K}]$. *This is an oracle algorithm that interacts with an attributes-specific decryption blackbox $\mathcal{D}$. By given the public parameter PP, an attribute set $S_{\mathcal{D}}$, and a probability value ϵ, the algorithm runs in time polynomial in λ and $1/\epsilon$, and outputs an index set $\mathbb{K}_T \subseteq [\mathcal{K}]$ which identifies the set of malicious users. Note that ϵ has to be polynomially related to λ, i.e. $\epsilon = 1/f(\lambda)$ for some polynomial f.*

The following tracing game captures the notion of **fully collusion-resistant traceability** against attributes-specific decryption blackbox. In the game, the adversary targets to build a decryption blackbox $\mathcal{D}$ that can decrypt ciphertexts generated under some attribute set $S_{\mathcal{D}}$ with non-negligible probability. The tracing algorithm, on the other side, is designed to extract the index of at least one of the malicious users whose decryption keys have been used for constructing $\mathcal{D}$.

$\mathsf{Game}_{\mathsf{TR}}$. The **tracing game** is defined between a challenger and an adversary $\mathcal{A}$ as follows:

Setup. The challenger runs $\mathsf{Setup}(\lambda, \mathcal{U}, \mathcal{K})$ and gives the public parameter PP to $\mathcal{A}$.
Key Query. For $i = 1$ to Q, $\mathcal{A}$ adaptively submits (index, access structure) pair $(k_i, \mathbb{A}_{k_i})$. The challenger responds with $\mathsf{SK}_{k_i, \mathbb{A}_{k_i}}$.
Decryption Blackbox Generation. $\mathcal{A}$ outputs a decryption blackbox $\mathcal{D}$ associated with an attribute set $S_{\mathcal{D}}$ and a non-negligible probability value ϵ.
Tracing. The challenger runs $\mathsf{Trace}^{\mathcal{D}}(\mathsf{PP}, S_{\mathcal{D}}, \epsilon)$ to obtain an index set $\mathbb{K}_T \subseteq [\mathcal{K}]$.

Let $\mathbb{K}_{\mathcal{D}} = \{k_i | 1 \leq i \leq Q\}$ be the index set of keys corrupted by the adversary. We say that the adversary $\mathcal{A}$ wins the game if the following conditions hold:

1. $\Pr[\mathcal{D}(\mathsf{Encrypt}(\mathsf{PP}, M, S_{\mathcal{D}})) = M] \geq \epsilon$, where the probability is taken over the random choices of message M and the random coins of $\mathcal{D}$. A decryption blackbox satisfying this condition is said to be a *useful attributes-specific decryption blackbox.*
2. $\mathbb{K}_T = \emptyset$, or $\mathbb{K}_T \not\subseteq \mathbb{K}_{\mathcal{D}}$, or ($S_{\mathcal{D}}$ *does not satisfy* $\mathbb{A}_{k_t}$ $\forall k_t \in \mathbb{K}_T$).

[4] Note that in the setting of predicate encryption [12], which can informally be regarded as a KP-ABE system with attribute-hiding property, the decryption blackbox [13] is also modeled similarly, i.e., the tracing algorithm takes as input an attribute I and a decryption blackbox $\mathcal{D}$ that decrypts ciphertexts associated with the attribute I.

We denote by $\mathsf{TRAdv}_{\mathcal{A}}$ the probability that adversary $\mathcal{A}$ wins this game.

Remark: For a useful attributes-specific decryption blackbox $\mathcal{D}$, the traced $\mathbb{K}_T$ must satisfy $(\mathbb{K}_T \neq \emptyset) \wedge (\mathbb{K}_T \subseteq \mathbb{K}_{\mathcal{D}}) \wedge (\exists k_t \in \mathbb{K}_T \; s.t. \; S_{\mathcal{D}} \; satisfies \; \mathbb{A}_{k_t})$ for traceabililty. (1) $(\mathbb{K}_T \neq \emptyset) \wedge (\mathbb{K}_T \subseteq \mathbb{K}_{\mathcal{D}})$ captures the preliminary traceability that the tracing algorithm can extract at least one malicious user and the coalition of malicious users cannot frame any innocent user. (2) $(\exists k_t \in \mathbb{K}_T \; s.t. \; S_{\mathcal{D}} \; satisfies \; \mathbb{A}_{k_t})$ captures *strong traceability* that the tracing algorithm can extract at least one malicious user whose private key enables $\mathcal{D}$ to have the decryption ability of decrypting ciphertexts generated under $S_{\mathcal{D}}$. We refer to [13,19] on why strong traceability is desirable.

Definition 3. *A $\mathcal{K}$-user KP-ABE scheme is traceable if for all PPT adversaries $\mathcal{A}$ the advantage $\mathsf{TRAdv}_{\mathcal{A}}$ is negligible in λ.*

We say that a $\mathcal{K}$-user KP-ABE scheme is *selectively traceable* if we add an **Init** stage before **Setup** where the adversary commits to the attribute set $S_{\mathcal{D}}$.

We emphasise that that we are modelling public traceability, namely, the Trace algorithm does not need any secrets and anyone can perform the tracing from the public parameters only. Also note that we are modelling a stateless (resettable) decryption blackbox – the decryption blackbox is just an oracle and maintains no state between activations.

3 Augmented KP-ABE

Following the routes of [19] where CP-ABE's traceability is discussed, instead of constructing a traceable KP-ABE directly, we define a simpler primitive called Augmented KP-ABE (or AugKP-ABE for short) and its security notions (message-hiding and index-hiding) first, then we show that an AugKP-ABE with message-hiding and index-hiding properties can be transformed to a secure KP-ABE with traceability. In Sec. 4, we propose a AugKP-ABE construction and prove its message-hiding and index-hiding properties in the standard model.

3.1 Definitions of Augmented KP-ABE

An Augmented KP-ABE (AugKP-ABE) has four algorithms: $\mathsf{Setup}_{\mathsf{A}}$, $\mathsf{KeyGen}_{\mathsf{A}}$, $\mathsf{Encrypt}_{\mathsf{A}}$, and $\mathsf{Decrypt}_{\mathsf{A}}$. The setup algorithm $\mathsf{Setup}_{\mathsf{A}}$ and key generation algorithm $\mathsf{KeyGen}_{\mathsf{A}}$ are the same as that of KP-ABE in Sec. 2.1. The encryption algorithm $\mathsf{Encrypt}_{\mathsf{A}}$ takes one more parameter $\bar{k} \in [\mathcal{K}+1]$ as input, and is defined as follows.

$\mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, S, \bar{k}) \rightarrow CT_S$. The algorithm takes as input PP, a message M, an attribute set $S \subseteq \mathcal{U}$, and an index $\bar{k} \in [\mathcal{K}+1]$, and outputs a ciphertext CT_S. S **is included in** CT_S**, but the value of** $\bar{k}$ **is not**.

The decryption algorithm $\mathsf{Decrypt}_{\mathsf{A}}$ is also defined in the same as that of the KP-ABE in Sec. 2.1. However, the correctness definition is changed to the following.

Correctness. for all access structures $\mathbb{A} \subseteq 2^{\mathcal{U}} \setminus \{\emptyset\}$, $k \in [\mathcal{K}]$, $S \subseteq \mathcal{U}$, $\bar{k} \in [\mathcal{K}+1]$, and messages M: if $(\mathsf{PP}, \mathsf{MSK}) \leftarrow \mathsf{Setup}_{\mathsf{A}}(\lambda, \mathcal{U}, \mathcal{K})$, $\mathsf{SK}_{k,\mathbb{A}} \leftarrow \mathsf{KeyGen}_{\mathsf{A}}(\mathsf{PP}, \mathsf{MSK}, \mathbb{A})$, $CT_S \leftarrow \mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, S, \bar{k})$, and $(S \; satisfies \; \mathbb{A}) \wedge (k \geq \bar{k})$, we have $\mathsf{Decrypt}_{\mathsf{A}}(\mathsf{PP}, CT, \mathsf{SK}_{k,\mathbb{A}}) = M$.

Remark: Note that during decryption, as long as S satisfies $\mathbb{A}$, the decryption algorithm outputs a message, but only when $k \geq \bar{k}$, the output message is equal to the correct message, that is, if

and only if (S satisfies $\mathbb{A}$) $\wedge$ ($k \geq \bar{k}$), can $\mathsf{SK}_{k,\mathbb{A}}$ correctly decrypt a ciphertext under $(S, \bar{k})$. If we always set $\bar{k} = 1$, the functions of AugKP-ABE are identical to that of KP-ABE. In fact, the idea behind transforming an AugKP-ABE to a blackbox traceable KP-ABE, that we will show shortly, is to construct an AugKP-ABE with index-hiding property, and then always sets $\bar{k} = 1$ in normal encryption, while using $\bar{k} \in [N+1]$ to generate ciphertexts for tracing.

Security. We define the security of AugKP-ABE in the following three games.

$\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$. The first game is a message-hiding game, denoted by $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$, is similar to $\mathsf{Game}_{\mathsf{MH}}$ except that the **Challenge** phase is

Challenge. $\mathcal{A}$ submits two equal-length messages M_0, M_1 and an attribute set S^*. The challenger flips a random coin $b \in \{0,1\}$, and sends $CT_{S^*} \leftarrow \mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M_b, S^*, 1)$ to $\mathcal{A}$.

$\mathcal{A}$ wins the game if $b' = b$ under the **restriction** that S^* cannot satisfy any of the queried access structures $\mathbb{A}_{k_1}, \ldots, \mathbb{A}_{k_Q}$. The advantage of $\mathcal{A}$ is defined as $\mathsf{MH}^{\mathsf{A}}_1\mathsf{Adv}_{\mathcal{A}} = |\Pr[b' = b] - \frac{1}{2}|$.

$\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$. The second game is also a message-hiding game, denoted by $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$, is similar to $\mathsf{Game}_{\mathsf{MH}}$ except that the **Challenge** phase is

Challenge. $\mathcal{A}$ submits two equal-length messages M_0, M_1 and an attribute set S^*. The challenger flips a random coin $b \in \{0,1\}$, and sends $CT_{S^*} \leftarrow \mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M_b, S^*, \mathcal{K}+1)$ to $\mathcal{A}$.

$\mathcal{A}$ wins the game if $b' = b$. The advantage of $\mathcal{A}$ is defined as $\mathsf{MH}^{\mathsf{A}}_1\mathsf{Adv}_{\mathcal{A}} = |\Pr[b' = b] - \frac{1}{2}|$.

Definition 4. *A $\mathcal{K}$-user Augmented KP-ABE scheme is message-hiding if for all PPT adversaries $\mathcal{A}$ the advantages $\mathsf{MH}^{\mathsf{A}}_1\mathsf{Adv}_{\mathcal{A}}$ and $\mathsf{MH}^{\mathsf{A}}_{\mathcal{K}+1}\mathsf{Adv}_{\mathcal{A}}$ are negligible functions of λ.*

$\mathsf{Game}^{\mathsf{A}}_{\mathsf{IH}}$. The third game, called **index hiding game**, requires that for any attribute set $S^* \subseteq \mathcal{U}$, no adversary can distinguish between an encryption using $(S^*, \bar{k})$ and one using $(S^*, \bar{k}+1)$ without a private key $\mathsf{SK}_{\bar{k},\mathbb{A}_{\bar{k}}}$ where S^* satisfies $\mathbb{A}_{\bar{k}}$. The game takes as input a parameter $\bar{k} \in [\mathcal{K}]$ which is given to both the challenger and the adversary $\mathcal{A}$. The game proceeds as follows:

Setup. The challenger runs $\mathsf{Setup}_{\mathsf{A}}(\lambda, \mathcal{U}, \mathcal{K})$ and sends PP to $\mathcal{A}$.

Key Query. For $i = 1$ to Q, $\mathcal{A}$ adaptively submits (index, access structure) pair $(k_i, \mathbb{A}_{k_i})$ to the challenger. The challenger responds with $\mathsf{SK}_{k_i,\mathbb{A}_{k_i}}$.

Challenge. $\mathcal{A}$ submits a message M and an attribute set S^*. The challenger flips a random coin $b \in \{0,1\}$, and sends $CT_{S^*} \leftarrow \mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, S^*, \bar{k}+b)$ to $\mathcal{A}$.

Guess. $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$ for b.

$\mathcal{A}$ wins the game if $b' = b$ under the **restriction** that none of the queried pairs $\{(k_i, \mathbb{A}_{k_i})\}$ can satisfy $(k_i = \bar{k}) \wedge (S^* \; satisfies \; \mathbb{A}_{k_i})$. The advantage of $\mathcal{A}$ is defined as $\mathsf{IH}^{\mathsf{A}}\mathsf{Adv}_{\mathcal{A}}[\bar{k}] = |\Pr[b' = b] - \frac{1}{2}|$.

Definition 5. *A $\mathcal{K}$-user Augmented KP-ABE scheme is index-hiding if for all PPT adversaries $\mathcal{A}$ the advantages $\mathsf{IH}^{\mathsf{A}}\mathsf{Adv}_{\mathcal{A}}[\bar{k}]$ for $\bar{k} = 1, \ldots, \mathcal{K}$ are negligible functions of λ.*

We say that an Augmented KP-ABE scheme is *selectively index-hiding* if we add an **Init** stage before **Setup** where the adversary commits to the challenge attribute set S^*.

3.2 Reducing Traceable KP-ABE to Augmented KP-ABE

Let $\Sigma_{\mathsf{A}} = (\mathsf{Setup}_{\mathsf{A}}, \mathsf{KeyGen}_{\mathsf{A}}, \mathsf{Encrypt}_{\mathsf{A}}, \mathsf{Decrypt}_{\mathsf{A}})$ be an AugKP-ABE, define $\mathsf{Encrypt}(\mathsf{PP}, M, \mathbb{A}) = \mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, \mathbb{A}, 1)$, then $\Sigma = (\mathsf{Setup}_{\mathsf{A}}, \mathsf{KeyGen}_{\mathsf{A}}, \mathsf{Encrypt}, \mathsf{Decrypt}_{\mathsf{A}})$ is a KP-ABE derived from Σ_{A}.

Theorem 1. *If Σ_{A} is message-hiding in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$, then Σ is secure.*

Proof. Note that Σ is a special case of Σ_{A} where the encryption algorithm always sets $\bar{k} = 1$. Hence, $\mathsf{Game}_{\mathsf{MH}}$ for Σ is identical to $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$ for Σ_{A}, which implies that $\mathsf{MHAdv}_{\mathcal{A}}$ for Σ in $\mathsf{Game}_{\mathsf{MH}}$ is equal to $\mathsf{MH}^{\mathsf{A}}_1\mathsf{Adv}_{\mathcal{A}}$ for Σ_{A} in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$, i.e., if Σ_{A} is message-hiding in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$, then Σ is secure.

Now we construct a tracing algorithm Trace for Σ and show that if Σ_{A} is message-hiding in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$ and (selectively) index-hiding, then Σ (equipped with Trace) is (selectively) traceable against attributes-specific decryption blackbox.

$\mathsf{Trace}^{\mathcal{D}}(\mathsf{PP}, S_{\mathcal{D}}, \epsilon) \rightarrow \mathbb{K}_T \subseteq [\mathcal{K}]$: Given an attribites-specific decryption blackbox $\mathcal{D}$ associated with an attribute set $S_{\mathcal{D}}$ and probability $\epsilon > 0$, the tracing algorithm works as follows: [5]

1. For $k = 1$ to $\mathcal{K} + 1$, do the following:
   (a) The algorithm repeats the following $8\lambda(\mathcal{K}/\epsilon)^2$ times:
      i. Sample M from the message space at random.
      ii. Let $CT_{S_{\mathcal{D}}} \leftarrow \mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, S_{\mathcal{D}}, k)$.
      iii. Query oracle $\mathcal{D}$ on input $CT_{S_{\mathcal{D}}}$, and compare the output of $\mathcal{D}$ with M.
   (b) Let $\hat{p}_k$ be the fraction of times that $\mathcal{D}$ decrypted the ciphertexts correctly.
2. Let $\mathbb{K}_T$ be the set of all $k \in [\mathcal{K}]$ for which $\hat{p}_k - \hat{p}_{k+1} \geq \epsilon/(4\mathcal{K})$. Then output $\mathbb{K}_T$ as the index set of the private decryption keys of malicious users.

Theorem 2. *If Σ_{A} is message-hiding in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$ and index-hiding (resp. selectively index-hiding), then Σ is traceable (resp. selectively traceable).*

Proof. In the proof sketch below, we show that if the attributes-specific decryption blackbox output by the adversary is a useful one then the traced $\mathbb{K}_T$ will satisfy $(\mathbb{K}_T \neq \emptyset) \wedge (\mathbb{K}_T \subseteq \mathbb{K}_{\mathcal{D}}) \wedge (\exists k_t \in \mathbb{K}_T \; s.t. \; S_{\mathcal{D}} \; satisfies \; \mathbb{A}_{k_t})$ with overwhelming probability, which implies that the adversary can win the game $\mathsf{Game}_{\mathsf{TR}}$ only with negligible probability, i.e., $\mathsf{TRAdv}_{\mathcal{A}}$ is negligible. The selective case is similar.

Let $\mathcal{D}$ be the attributes-specific decryption blackbox output by the adversary, and $S_{\mathcal{D}}$ be the attribute set describing $\mathcal{D}$. Define

$$p_{\bar{k}} = \Pr[\mathcal{D}(\mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, S_{\mathcal{D}}, \bar{k})) = M],$$

where the probability is taken over the random choice of message M and the random coins of $\mathcal{D}$. We have that $p_1 \geq \epsilon$ and $p_{\mathcal{K}+1}$ is negligible (for simplicity let $p_{\mathcal{K}+1} = 0$). The former follows from the fact that $\mathcal{D}$ is useful, and the latter is because Σ_{A} is message-hiding in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$. Then there must exist some $k \in [\mathcal{K}]$ such that $p_k - p_{k+1} \geq \epsilon/(2\mathcal{K})$. By the Chernoff bound it follows that with overwhelming probability, $\hat{p}_k - \hat{p}_{k+1} \geq \epsilon/(4\mathcal{K})$. Hence, we have $\mathbb{K}_T \neq \emptyset$.

For any $k \in \mathbb{K}_T$ (i.e., $\hat{p}_k - \hat{p}_{k+1} \geq \frac{\epsilon}{4\mathcal{K}}$), we know, by Chernoff, that with overwhelming probability $p_k - p_{k+1} \geq \epsilon/(8\mathcal{K})$. Clearly $(k \in \mathbb{K}_{\mathcal{D}}) \wedge (S_{\mathcal{D}} \; satisfies \; \mathbb{A}_k)$ since otherwise, $\mathcal{D}$ can be directly used to win the index-hiding game for Σ_A. Hence, we have $(\mathbb{K}_T \subseteq \mathbb{K}_{\mathcal{D}}) \wedge (S_{\mathcal{D}} \; satisfies \; \mathbb{A}_k \; \forall k \in \mathbb{K}_T)$.

[5] The tracing algorithm uses a technique based on that in broadcast encryption by [4,5,8].

4 An Efficient Augmented KP-ABE Scheme

We now propose an AugKP-ABE scheme which can be considered as combining the KP-ABE scheme of [14] and the traitor tracing scheme of [8]. We stress that this work is not a trivial combination of the two schemes, which may result in insecure or inefficient schemes, as discussed in [19]. The proposed AugKP-ABE scheme is highly expressive in supporting any monotonic access structures, and is efficient with ciphertext size $O(\sqrt{\mathcal{K}} + |S|)$, where $\mathcal{K}$ is the number of users in the system and S is the attribute set of the ciphertext. We prove that the scheme is adaptively message-hiding and selectively index-hiding in the standard model. Combining this AugKP-ABE scheme with the result in Sec. 3.2, we obtain a fully secure and highly expressive KP-ABE scheme which is simultaneously selectively traceable, and for a fully collusion-resistant blackbox traceable system the resulting KP-ABE scheme achieves the most efficient level to date, with overhead linear in $\sqrt{\mathcal{K}}$.

4.1 Preliminaries

Linear Secret-Sharing Schemes. As shown in [2], any monotonic access structure can be realized by a linear secret sharing scheme.

Definition 6. (Linear Secret-Sharing Schemes (LSSS)) *[26] A secret sharing scheme Π over attribute universe $\mathcal{U}$ is called linear (over $\mathbb{Z}_p$) if*

1. *The shares for each attribute form a vector over $\mathbb{Z}_p$.*
2. *There exists a matrix A called the share-generating matrix for Π. The matrix A has l rows and n columns. For $i = 1, \ldots, l$, the i^{th} row A_i of A is labeled by an attribute $\rho(i)$(ρ is a function from $\{1, \ldots, l\}$ to $\mathcal{U}$). When we consider the column vector $\boldsymbol{v} = (s, r_2, \ldots, r_n)$, where $s \in \mathbb{Z}_p$ is the secret to be shared and $r_2, \ldots, r_n \in \mathbb{Z}_p$ are randomly chosen, then $A\boldsymbol{v}$ is the vector of l shares of the secret s according to Π. The share $\lambda_i = (A\boldsymbol{v})_i$, i.e., the inner product $A_i \cdot \boldsymbol{v}$, belongs to attribute $\rho(i)$.*

Also shown in [2], every LSSS as defined above enjoys the linear reconstruction property, which is defined as follows: Suppose that Π is an LSSS for access structure $\mathbb{A}$. Let $S \in \mathbb{A}$ be an authorized set, and $I \subset \{1, \ldots, l\}$ be defined as $I = \{i : \rho(i) \in S\}$. There exist constants $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ such that if $\{\lambda_i\}$ are valid shares of a secret s according to Π, $\sum_{i \in I} \omega_i \lambda_i = s$. Furthermore, these constants $\{\omega_i\}$ can be found in time polynomial in the size of the share-generating matrix A. For any unauthorized set, no such constants exist. In this paper, as of previous work, we use an LSSS matrix (A, ρ) to express an access structure associated to a private decryption key.

Composite Order Bilinear Groups [3]. Let $\mathcal{G}$ be a group generator algorithm, which takes a security parameter λ and outputs $(p_1, p_2, p_3, \mathbb{G}, \mathbb{G}_T, e)$ where p_1, p_2, p_3 are distinct primes, $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of order $N = p_1 p_2 p_3$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a map such that: (1) (Bilinear) $\forall g, h \in \mathbb{G}, a, b \in \mathbb{Z}_N, e(g^a, h^b) = e(g, h)^{ab}$, (2) (Non-Degenerate) $\exists g \in \mathbb{G}$ such that $e(g, g)$ has order N in $\mathbb{G}_T$. Assume that group operations in $\mathbb{G}$ and $\mathbb{G}_T$ as well as the bilinear map e are computable in polynomial time with respect to λ. Let $\mathbb{G}_{p_1}$, $\mathbb{G}_{p_2}$ and $\mathbb{G}_{p_3}$ be the subgroups of order p_1, p_2 and p_3 in $\mathbb{G}$ respectively. These subgroups are "orthogonal" to each other under the bilinear map e: if $h_i \in \mathbb{G}_{p_i}$ and $h_j \in \mathbb{G}_{p_j}$ for $i \neq j$, $e(h_i, h_j) = 1$ (the identity element in $\mathbb{G}_T$).

Complexity Assumptions. The message-hiding property of our AugKP-ABE scheme will be based on three assumptions (Assumption 1, 2 and 3 in [15]) that are used by [15,14] to achieve full

security of their ABE schemes, and the index-hiding property will be based on two assumptions (Decision 3-Party Diffie-Hellman Assumption and the Decisional Linear Assumption) that are used by [8] to achieve traceability in the setting of broadcast encryption. We refer to [15,8] for the details of these assumptions.

Notations. Suppose the number of users $\mathcal{K}$ in the system equals m^2 for some m.[6] We arrange the users in an $m \times m$ matrix and uniquely assign a tuple (i, j) where $1 \leq i, j \leq m$, to each user. A user at position (i, j) of the matrix has index $k = (i-1) * m + j$. For simplicity, we directly use (i, j) as the index where $(i, j) \geq (\bar{i}, \bar{j})$ means that $((i > \bar{i}) \vee (i = \bar{i} \wedge j \geq \bar{j}))$. The use of pairwise notation (i, j) is purely a notational convenience, as $k = (i-1) * m + j$ defines a bijection between $\{(i, j) | 1 \leq i, j \leq m\}$ and $[\mathcal{K}]$. For a given vector $\boldsymbol{v} = (v_1, \ldots, v_d)$, by $g^{\boldsymbol{v}}$ we mean the vector $(g^{v_1}, \ldots, g^{v_d})$. Furthermore, for $g^{\boldsymbol{v}} = (g^{v_1}, \ldots, g^{v_d})$ and $g^{\boldsymbol{w}} = (g^{w_1}, \ldots, g^{w_d})$, by $g^{\boldsymbol{v}} \cdot g^{\boldsymbol{w}}$ we mean the vector $(g^{v_1+w_1}, \ldots, g^{v_d+w_d})$, i.e. $g^{\boldsymbol{v}} \cdot g^{\boldsymbol{w}} = g^{\boldsymbol{v}+\boldsymbol{w}}$, and by $e_d(g^{\boldsymbol{v}}, g^{\boldsymbol{w}})$ we mean $\prod_{k=1}^{d} e(g^{v_k}, g^{w_k})$, i.e. $e_d(g^{\boldsymbol{v}}, g^{\boldsymbol{w}}) = \prod_{k=1}^{d} e(g^{v_k}, g^{w_k}) = e(g, g)^{(\boldsymbol{v} \cdot \boldsymbol{w})}$ where $(\boldsymbol{v} \cdot \boldsymbol{w})$ is the inner product of $\boldsymbol{v}$ and $\boldsymbol{w}$. Given a bilinear group order N, one can randomly choose $r_x, r_y, r_z \in \mathbb{Z}_N$, and set $\boldsymbol{\chi}_1 = (r_x, 0, r_z)$, $\boldsymbol{\chi}_2 = (0, r_y, r_z)$, $\boldsymbol{\chi}_3 = \boldsymbol{\chi}_1 \times \boldsymbol{\chi}_2 = (-r_y r_z, -r_x r_z, r_x r_y)$. Let $span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\}$ be the subspace spanned by $\boldsymbol{\chi}_1$ and $\boldsymbol{\chi}_2$, i.e. $span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\} = \{\nu_1 \boldsymbol{\chi}_1 + \nu_2 \boldsymbol{\chi}_2 | \nu_1, \nu_2 \in \mathbb{Z}_N\}$. We can see that $\boldsymbol{\chi}_3$ is orthogonal to the subspace $span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\}$ and $\mathbb{Z}_N^3 = span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \boldsymbol{\chi}_3\} = \{\nu_1 \boldsymbol{\chi}_1 + \nu_2 \boldsymbol{\chi}_2 + \nu_3 \boldsymbol{\chi}_3 | \nu_1, \nu_2, \nu_3 \in \mathbb{Z}_N\}$. For any $\boldsymbol{v} \in span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\}$, we have $(\boldsymbol{\chi}_3 \cdot \boldsymbol{v}) = 0$, and for random $\boldsymbol{v} \in \mathbb{Z}_N^3$, $(\boldsymbol{\chi}_3 \cdot \boldsymbol{v}) \neq 0$ happens with overwhelming probability.

4.2 AugKP-ABE Construction

$\mathsf{Setup}_{\mathsf{A}}(\lambda, \mathcal{U}, \mathcal{K} = m^2) \rightarrow (\mathsf{PP}, \mathsf{MSK})$. Let $\mathbb{G}$ be a bilinear group of order $N = p_1 p_2 p_3$ (3 distinct primes, whose size is determined by λ), $\mathbb{G}_{p_i}$ the subgroup of order p_i in $\mathbb{G}$ (for $i = 1, 2, 3$), and $g, f \in \mathbb{G}_{p_1}$, $g_3 \in \mathbb{G}_{p_3}$ the generators of corresponding subgroups. The algorithm randomly chooses exponents $\alpha \in \mathbb{Z}_N$, $\{\alpha_i,\ r_i,\ z_i \in \mathbb{Z}_N\}_{i \in [m]}$, $\{c_j \in \mathbb{Z}_N\}_{j \in [m]}$, $\{a_x \in \mathbb{Z}_N\}_{x \in \mathcal{U}}$. The public parameter PP includes the description of the group and the following elements:

$$\Big(g, f,\ E = e(g, g)^{\alpha},\ \{E_i = e(g, g)^{\alpha_i},\ G_i = g^{r_i},\ Z_i = g^{z_i}\}_{i \in [m]},\ \{H_j = g^{c_j}\}_{j \in [m]},\ \{U_x = g^{a_x}\}_{x \in \mathcal{U}}\Big).$$

The master secret key is set to $\mathsf{MSK} = (\alpha,\ \alpha_1, \ldots, \alpha_m,\ r_1, \ldots, r_m,\ c_1, \ldots, c_m,\ g_3\ )$. In addition, a counter $ctr = 0$ is included in MSK.

$\mathsf{KeyGen}_{\mathsf{A}}(\mathsf{PP}, \mathsf{MSK}, (A, \rho)) \rightarrow \mathsf{SK}_{(i,j),(A,\rho)}$. A is an $l \times n$ LSSS matrix and ρ maps each row A_k of A to an attribute $\rho(k) \in \mathcal{U}$. It is required that ρ would not map two different rows to the same attribute[7]. The algorithm first sets $ctr = ctr + 1$ and computes the corresponding index in the form of (i, j) where $1 \leq i, j \leq m$ and $(i-1) * m + j = ctr$. Then it randomly chooses $\boldsymbol{u} = (\sigma_{i,j}, u_2, \ldots, u_n) \in \mathbb{Z}_N^n$, $w_2, \ldots, w_n \in \mathbb{Z}_N$, and $\{\xi_k \in \mathbb{Z}_N, R_{k,1}, R_{k,2} \in \mathbb{G}_{p_3}\}_{k \in [l]}$. Let $\boldsymbol{w} = (\alpha, w_2, \ldots, w_n)$, the algorithms outputs a private key $\mathsf{SK}_{(i,j),(A,\rho)} = \big((i, j), (A, \rho),\ K_{i,j}, K'_{i,j}, K''_{i,j},\ \{K_{i,j,k,1},\ K_{i,j,k,2}\}_{k \in [l]}\big)$ where

$$K_{i,j} = g^{\alpha_i} g^{r_i c_j} f^{\sigma_{i,j}},\ K'_{i,j} = g^{\sigma_{i,j}},\ K''_{i,j} = Z_i^{\sigma_{i,j}},$$

[6] If the number of users is not a square, we add some "dummy" users to pad to the next square.

[7] This restriction is inherited from the underlying KP-ABE scheme [14], and can be removed with the techniques in [14] similarly, with some loss of efficiency. The similar restriction in CP-ABE has been efficiently eliminated recently by Lewko and Waters in [16], but fully secure KP-ABE scheme without this restriction is not proposed yet.

$$\{K_{i,j,k,1} = f^{(A_k \cdot \boldsymbol{u})} g^{(A_k \cdot \boldsymbol{w})} U_{\rho(k)}^{\xi_k} R_{k,1},\ K_{i,j,k,2} = g^{\xi_k} R_{k,2}\}_{k \in [l]}.$$

$\mathsf{Encrypt}_{\mathsf{A}}(\mathsf{PP}, M, S, (\bar{i}, \bar{j})) \rightarrow CT_S$. The algorithm randomly chooses

$$\pi,\ \kappa,\ \tau,\ s_1, \ldots, s_m,\ t_1, \ldots, t_m \in \mathbb{Z}_N,$$
$$\boldsymbol{v}_c,\ \boldsymbol{w}_1, \ldots, \boldsymbol{w}_m \in \mathbb{Z}_N^3.$$

In addition, the algorithm randomly chooses $r_x, r_y, r_z \in \mathbb{Z}_N$, and sets $\boldsymbol{\chi}_1 = (r_x, 0, r_z)$, $\boldsymbol{\chi}_2 = (0, r_y, r_z)$, $\boldsymbol{\chi}_3 = \boldsymbol{\chi}_1 \times \boldsymbol{\chi}_2 = (-r_y r_z, -r_x r_z, r_x r_y)$. Then it randomly chooses

$$\boldsymbol{v}_i \in \mathbb{Z}_N^3\ \forall i \in \{1, \ldots, \bar{i}\},$$
$$\boldsymbol{v}_i \in span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\}\ \forall i \in \{\bar{i}+1, \ldots, m\},$$

and creates the ciphertext $\langle S,\ (\boldsymbol{R}_i, \boldsymbol{R}'_i, Q_i, Q'_i, Q''_i, T_i)_{i=1}^m,\ (\boldsymbol{C}_j, \boldsymbol{C}'_j)_{j=1}^m,\ P, \{P_x\}_{x \in S}\rangle$ as follows:

1. For each $i \in [m]$:
   - if $i < \bar{i}$: it randomly chooses $\hat{s}_i \in \mathbb{Z}_N$, and sets

$$\boldsymbol{R}_i = g^{\boldsymbol{v}_i},\ \ \boldsymbol{R}'_i = g^{\kappa \boldsymbol{v}_i},\qquad Q_i = g^{s_i},\ \ Q'_i = f^{s_i} Z_i^{t_i} f^{\pi},\ \ Q''_i = g^{t_i},\qquad T_i = E_i^{\hat{s}_i}.$$

   - if $i \geq \bar{i}$: it sets

$$\boldsymbol{R}_i = G_i^{s_i \boldsymbol{v}_i},\ \ \boldsymbol{R}'_i = G_i^{\kappa s_i \boldsymbol{v}_i},$$
$$Q_i = g^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)},\ \ Q'_i = f^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} Z_i^{t_i} f^{\pi},\ \ Q''_i = g^{t_i},\qquad T_i = M \cdot E_i^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} \cdot E^{\pi}.$$

2. For each $j \in [m]$:
   - if $j < \bar{j}$: it randomly chooses $\mu_j \in \mathbb{Z}_N$, and sets $\boldsymbol{C}_j = H_j^{\tau(\boldsymbol{v}_c + \mu_j \boldsymbol{\chi}_3)} \cdot g^{\kappa \boldsymbol{w}_j},\ \ \boldsymbol{C}'_j = g^{\boldsymbol{w}_j}$.
   - if $j \geq \bar{j}$: it sets $\boldsymbol{C}_j = H_j^{\tau \boldsymbol{v}_c} \cdot g^{\kappa \boldsymbol{w}_j},\ \ \boldsymbol{C}'_j = g^{\boldsymbol{w}_j}$.
3. It sets $P = g^{\pi},\ \ \{P_x = U_x^{\pi}\}_{x \in S}$.

$\mathsf{Decrypt}_{\mathsf{A}}(\mathsf{PP}, CT_S, \mathsf{SK}_{(i,j),(A,\rho)}) \rightarrow M$ or $\perp$. For ciphertext $CT_S = \langle S, (\boldsymbol{R}_i, \boldsymbol{R}'_i, Q_i, Q'_i, Q''_i, T_i)_{i=1}^m, (\boldsymbol{C}_j, \boldsymbol{C}'_j)_{j=1}^m, P, \{P_x\}_{x \in S}\rangle$ and secret decryption key $\mathsf{SK}_{(i,j),(A,\rho)} = \big((i,j), (A,\rho),\ K_{i,j}, K'_{i,j}, K''_{i,j}, \{K_{i,j,k,1},\ K_{i,j,k,2}\}_{k \in [l]}\big)$, if S does not satisfy (A, ρ), the algorithm outputs $\perp$, otherwise it

1. computes constants $\{\omega_k \in \mathbb{Z}_N\}$ such that $\sum_{\rho(k) \in S} \omega_k A_k = (1, 0, \ldots, 0)$, then computes

$$D_P = \prod_{\rho(k) \in S} \Big(\frac{e(K_{i,j,k,1}, P)}{e(K_{i,j,k,2}, P_{\rho(k)})}\Big)^{\omega_k} = \prod_{\rho(k) \in S} \big(e(f^{(A_k \cdot \boldsymbol{u})} g^{(A_k \cdot \boldsymbol{w})}, g^{\pi})\big)^{\omega_k} = e(f, g)^{\pi \sigma_{i,j}} e(g, g)^{\alpha \pi};$$

2. computes

$$D_I = \frac{e(K_{i,j}, Q_i) \cdot e(K''_{i,j}, Q''_i)}{e(K'_{i,j}, Q'_i)} \cdot \frac{e_3(\boldsymbol{R}'_i, \boldsymbol{C}'_j)}{e_3(\boldsymbol{R}_i, \boldsymbol{C}_j)}.$$

3. computes $M' = T_i / (D_P \cdot D_I)$ as the output message.

Correctness. Assume the message is M and the encryption index is $(\bar{i}, \bar{j})$, Appendix A shows that

$$D_I = \begin{cases} E_i^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} / e(g, f)^{\pi \sigma_{i,j}}, & : (i > \bar{i})\ or\ (i = \bar{i} \wedge j \geq \bar{j}) \\ E_i^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} / \big(e(g, f)^{\pi \sigma_{i,j}} e(g, g)^{r_i s_i c_j \tau \mu_j (\boldsymbol{v}_i \cdot \boldsymbol{\chi}_3)}\big), & : (i = \bar{i} \wedge j < \bar{j}). \end{cases}$$

Thus, we have (1) if $(i > \bar{i}) \vee (i = \bar{i} \wedge j \geq \bar{j})$, then $M' = M$; (2) if $i = \bar{i} \wedge j < \bar{j}$, then $M' = M \cdot e(g, g)^{\tau s_i r_i c_j \mu_j (\boldsymbol{v}_i \cdot \boldsymbol{\chi}_3)}$; (3) if $i < \bar{i}$, then M' has no relation with M.

4.3 AugKP-ABE Security

The following Theorem 3 and 4 show that our AugKP-ABE construction in Sec. 4.2 is message-hiding, and Theorem 5 shows that our construction is selectively index-hiding.

Theorem 3. *Suppose that Assumptions 1, 2, and 3 in [15] hold. Then no PPT adversary can win* $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$ *with non-negligible advantage.*

Proof. The structure of the KP-ABE portion of our AugKP-ABE is similar to that of the KP-ABE in [14], the proof of Theorem 3 is also similar to that of [14]. Here we prove the theorem by reducing the message-hiding property of our AugKP-ABE scheme in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$ to the security of the KP-ABE in [14]. The proof is given in Appendix B.1.

Theorem 4. *No PPT adversary can win* $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$ *with non-negligible advantage.*

Proof. The argument for the message-hiding property in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_{\mathcal{K}+1}}$ is very straightforward since an encryption to index $\mathcal{K}+1 = (m+1, 1)$ contains no information about the message. The simulator simply runs actual $\mathsf{Setup}_{\mathsf{A}}$ and $\mathsf{KeyGen}_{\mathsf{A}}$ algorithms and encrypts the message M_b by the challenge attribute set S^* and index $(m+1, 1)$. Since for all $i = 1$ to m, the values of $T_i = e(g, g)^{\hat{s}_i}$ contains no information about the message, the bit b is perfectly hidden and $\mathsf{MH}^{\mathsf{A}}_{\mathcal{K}+1}\mathsf{Adv}_{\mathcal{A}} = 0$.

Theorem 5. *Suppose that the Decision 3-Party Diffie-Hellman Assumption and the Decisional Linear Assumption hold (referring to [8] for the details of the two assumptions). Then no PPT adversary can selectively win* $\mathsf{Game}^{\mathsf{A}}_{\mathsf{IH}}$ *with non-negligible advantage.*

Proof. Theorem 5 follows Lemma 1 and Lemma 2 below.

Lemma 1. *If the Decision 3-Party Diffie-Hellman Assumption holds, then for* $\bar{j} < m$ *no PPT adversary can selectively distinguish between an encryption to* $(\bar{i}, \bar{j})$ *and an encryption to* $(\bar{i}, \bar{j}+1)$ *in* $\mathsf{Game}^{\mathsf{A}}_{\mathsf{IH}}$ *with non-negligible advantage.*

Proof. The proof of this lemma explores the techniques of combining a traitor tracing scheme and a KP-ABE scheme. When the adversary queries a private key with the index $(\bar{i}, \bar{j})$, the game restriction implies that the corresponding access structure must not be satisfied by the challenge attribute set S^*. In other words, we have to use a restriction on "attributes and access structure" to prove the index-hiding property on "index", which are very uncorrelated structures. The ciphertext components $Z_i^{t_i}$ (in Q_i') and $Q_i'' = g^{t_i}$ works like a "transmission gear" to intertwine the two structures, *securely* combining the tracing part ($f^{\tau s_i(\boldsymbol{v}_i \cdot \boldsymbol{v}_c)}$ for $i \geq \bar{i}$ and f^{s_i} for $i < \bar{i}$) and the KP-ABE part (f^{π}) together. The proof is given in Appendix B.2.

Lemma 2. *Suppose that the Decision 3-Party Diffie-Hellman Assumption and the Decisional Linear Assumption hold. Then for* $1 \leq \bar{i} \leq m$ *no PPT adversary can selectively distinguish between an encryption to* $(\bar{i}, m)$ *and one to* $(\bar{i}+1, 1)$ *in* $\mathsf{Game}^{\mathsf{A}}_{\mathsf{IH}}$ *with non-negligible advantage.*

Proof. Similar to the proof of Lemma 6.3 in [8], to prove this lemma we define the following hybrid experiments: H_1: Encrypt to $(\bar{i}, \bar{j} = m)$; H_2: Encrypt to $(\bar{i}, \bar{j} = m+1)$; and H_3: Encrypt to $(\bar{i}+1, 1)$. Lemma 2 follows Claim 1 and Claim 2 below.

Claim 1. *If the Decision 3-Party Diffie-Hellman Assumption holds, then no PPT adversary can selectively distinguish between experiment* H_1 *and* H_2 *with non-negligible advantage.*

Proof. The proof is identical to that of Lemma 1.

Claim 2. *Suppose that the Decision 3-Party Diffie-Hellman Assumption and the Decisional Linear Assumption hold. Then no PPT adversary can distinguish between experiments H_2 and H_3 with non-negligible advantage.*

Proof. The indistinguishability of H_2 and H_3 can be proven using the similar proof to that of Lemma 6.3 in [8], which was used to prove the indistinguishability of similar hybrid experiments for their Augmented Broadcast Encryption (AugBE) scheme. We will prove Claim 2 by a reduction between our AugKP-ABE scheme and the AugBE scheme in [8, Sec.5.1]. The proof is given in Appendix B.3.

5 Conclusion

We proposed an expressive and efficient KP-ABE scheme that simultaneously supports fully collusion-resistant (and public) blackbox traceability and high expressiveness (i.e. supporting any monotonic access structures). The scheme is proven fully secure in the standard model and selectively traceable in the standard model. Compared with the most efficient conventional (non-traceable) KP-ABE schemes in the literature with high expressiveness and full security, our scheme adds fully collusion-resistant blackbox traceability with the price of adding only $O(\sqrt{\mathcal{K}})$ elements in the ciphertext and public key. Instead of directly building a traceable KP-ABE scheme, we constructed a simpler primitive called Augmented KP-ABE, and showed that an Augmented KP-ABE scheme with message-hiding and index-hiding properties is sufficient for constructing a secure KP-ABE scheme with fully collusion-resistant blackbox traceability.

References

1. Attrapadung, N., Libert, B., de Panafieu, E.: Expressive key-policy attribute-based encryption with constant-size ciphertexts. In: Public Key Cryptography. pp. 90–108 (2011)
2. Beimel, A.: Secure Schemes for Secret Sharing and Key Distribution. Ph.D. thesis, Israel Institute of Technology, Technion, Haifa, Israel (1996)
3. Boneh, D., Goh, E.J., Nissim, K.: Evaluating 2-dnf formulas on ciphertexts. In: TCC. pp. 325–341 (2005)
4. Boneh, D., Sahai, A., Waters, B.: Fully collusion resistant traitor tracing with short ciphertexts and private keys. In: EUROCRYPT. pp. 573–592 (2006)
5. Boneh, D., Waters, B.: A fully collusion resistant broadcast, trace, and revoke system. In: ACM Conference on Computer and Communications Security. pp. 211–220 (2006)
6. Cheung, L., Newport, C.C.: Provably secure ciphertext policy abe. In: ACM Conference on Computer and Communications Security. pp. 456–465 (2007)
7. Garg, S., Gentry, C., Halevi, S., Sahai, A., Waters, B.: Attribute-based encryption for circuits from multilinear maps. In: CRYPTO (2). pp. 479–499 (2013)
8. Garg, S., Kumarasubramanian, A., Sahai, A., Waters, B.: Building efficient fully collusion-resilient traitor tracing and revocation schemes. In: ACM Conference on Computer and Communications Security. pp. 121–130 (2010)
9. Goyal, V., Jain, A., Pandey, O., Sahai, A.: Bounded ciphertext policy attribute based encryption. In: ICALP (2). pp. 579–591 (2008)
10. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: ACM Conference on Computer and Communications Security. pp. 89–98 (2006)
11. Herranz, J., Laguillaumie, F., Ràfols, C.: Constant size ciphertexts in threshold attribute-based encryption. In: Public Key Cryptography. pp. 19–34 (2010)
12. Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: EUROCRYPT. pp. 146–162 (2008)

13. Katz, J., Schröder, D.: Tracing insider attacks in the context of predicate encryption schemes. In: ACITA (2011), https://www.usukita.org/node/1779
14. Lewko, A.B., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. IACR Cryptology ePrint Archive 2010, 110 (2010)
15. Lewko, A.B., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In: EUROCRYPT. pp. 62–91 (2010)
16. Lewko, A.B., Waters, B.: New proof methods for attribute-based encryption: Achieving full security through selective techniques. In: CRYPTO. pp. 180–198 (2012)
17. Li, J., Huang, Q., Chen, X., Chow, S.S.M., Wong, D.S., Xie, D.: Multi-authority ciphertext-policy attribute-based encryption with accountability. In: ASIACCS. pp. 386–390 (2011)
18. Li, J., Ren, K., Kim, K.: A2be: Accountable attribute-based encryption for abuse free access control. IACR Cryptology ePrint Archive 2009, 118 (2009)
19. Liu, Z., Cao, Z., Wong, D.S.: Blackbox traceable CP-ABE: how to catch people leaking their keys by selling decryption devices on ebay. In: ACM Conference on Computer and Communications Security. pp. 475–486 (2013)
20. Liu, Z., Cao, Z., Wong, D.S.: White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures. IEEE Transactions on Information Forensics and Security 8(1), 76–88 (2013)
21. Okamoto, T., Takashima, K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: CRYPTO. pp. 191–208 (2010)
22. Ostrovsky, R., Sahai, A., Waters, B.: Attribute-based encryption with non-monotonic access structures. In: ACM Conference on Computer and Communications Security. pp. 195–203 (2007)
23. Rouselakis, Y., Waters, B.: Practical constructions and new proof methods for large universe attribute-based encryption. In: ACM Conference on Computer and Communications Security. pp. 463–474 (2013)
24. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: EUROCRYPT. pp. 457–473 (2005)
25. Wang, Y.T., Chen, K.F., Chen, J.H.: Attribute-based traitor tracing. J. Inf. Sci. Eng. 27(1), 181–195 (2011)
26. Waters, B.: Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In: Public Key Cryptography. pp. 53–70 (2011)
27. Waters, B.: Functional encryption for regular languages. In: CRYPTO. pp. 218–235 (2012)
28. Yu, S., Ren, K., Lou, W., Li, J.: Defending against key abuse attacks in KP-ABE enabled broadcast systems. In: SecureComm. pp. 311–329 (2009)

A Correctness of Our AugKP-ABE Construction

Correctness. Assume the encryption index is $(\bar{i}, \bar{j})$. Note that for $i \geq \bar{i}$ we have

$$\frac{e(K_{i,j}, Q_i) \cdot e(K''_{i,j}, Q''_i)}{e(K'_{i,j}, Q'_i)} = \frac{e(g^{\alpha_i} g^{r_i c_j} f^{\sigma_{i,j}}, g^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)}) e(Z_i^{\sigma_{i,j}}, g^{t_i})}{e(g^{\sigma_{i,j}}, f^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} Z_i^{t_i} f^{\pi})} = \frac{e(g^{\alpha_i}, g^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)}) e(g^{r_i c_j}, g^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)})}{e(g^{\sigma_{i,j}}, f^{\pi})},$$

- if $i \geq \bar{i} \wedge j \geq \bar{j}$: $\frac{e_3(\boldsymbol{R}'_i, \boldsymbol{C}'_j)}{e_3(\boldsymbol{R}_i, \boldsymbol{C}_j)} = \frac{e_3(G_i^{\kappa s_i \boldsymbol{v}_i}, g^{\boldsymbol{w}_j})}{e_3(G_i^{s_i \boldsymbol{v}_i}, H_j^{\tau \boldsymbol{v}_c} \cdot g^{\kappa \boldsymbol{w}_j})} = \frac{1}{e_3(g^{r_i s_i \boldsymbol{v}_i}, g^{c_j \tau \boldsymbol{v}_c})} = \frac{1}{e(g,g)^{r_i s_i c_j \tau (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)}}$,
- if $i > \bar{i} \wedge j < \bar{j}$: since $\boldsymbol{v}_i \in span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\}$, we have $(\boldsymbol{v}_i \cdot \boldsymbol{\chi}_3) = 0$. Then

$$\frac{e_3(\boldsymbol{R}'_i, \boldsymbol{C}'_j)}{e_3(\boldsymbol{R}_i, \boldsymbol{C}_j)} = \frac{e_3(G_i^{\kappa s_i \boldsymbol{v}_i}, g^{\boldsymbol{w}_j})}{e_3(G_i^{s_i \boldsymbol{v}_i}, H_j^{\tau(\boldsymbol{v}_c + \mu_j \boldsymbol{\chi}_3)} \cdot g^{\kappa \boldsymbol{w}_j})} = \frac{1}{e_3(g^{r_i s_i \boldsymbol{v}_i}, g^{c_j \tau (\boldsymbol{v}_c + \mu_j \boldsymbol{\chi}_3)})} = \frac{1}{e(g,g)^{r_i s_i c_j \tau (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)}},$$

- if $i = \bar{i} \wedge j < \bar{j}$: since $\boldsymbol{v}_i$ is randomly chosen from $\mathbb{Z}_N^3$ (resp. $\mathbb{Z}_p^3$), we have that $(\boldsymbol{v}_i \cdot \boldsymbol{\chi}_3) \neq 0$ happens with overwhelming probability. Then

$$\begin{aligned}\frac{e_3(\boldsymbol{R}'_i, \boldsymbol{C}'_j)}{e_3(\boldsymbol{R}_i, \boldsymbol{C}_j)} &= \frac{e_3(G_i^{\kappa s_i \boldsymbol{v}_i}, g^{\boldsymbol{w}_j})}{e_3(G_i^{s_i \boldsymbol{v}_i}, H_j^{\tau(\boldsymbol{v}_c + \mu_j \boldsymbol{\chi}_3)} \cdot g^{\kappa \boldsymbol{w}_j})} = \frac{1}{e_3(g^{r_i s_i \boldsymbol{v}_i}, g^{c_j \tau (\boldsymbol{v}_c + \mu_j \boldsymbol{\chi}_3)})} \\ &= \frac{1}{e(g,g)^{r_i s_i c_j \tau (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} e(g,g)^{r_i s_i c_j \tau \mu_j (\boldsymbol{v}_i \cdot \boldsymbol{\chi}_3)}}.\end{aligned}$$

B Proofs

B.1 Proof of Theorem 3

Proof. The Theorem 33 of [14] states that *If Assumptions 1, 2, and 3 in [15] hold, then the KP-ABE scheme in [14] is secure.* To prove the Theorem 3, we do not build a direct reduction to the underlying assumptions, instead, we build a reduction to attack the KP-ABE scheme in [14]. In particular, suppose there is a PPT adversary $\mathcal{A}$ that can break our AugKP-ABE scheme Σ_{A} in $\mathsf{Game}^{\mathsf{A}}_{\mathsf{MH}_1}$ with non-negligible advantage $\mathsf{MH}_1^{\mathsf{A}}\mathsf{Adv}_{\mathcal{A}}$, we construct a PPT algorithm $\mathcal{B}$ to break the KP-ABE scheme (denoted by Σ_{kpabe}) in [14] with advantage $Adv_{\mathcal{B}}\Sigma_{\mathsf{kpabe}}$, which equals to $\mathsf{MH}_1^{\mathsf{A}}\mathsf{Adv}_{\mathcal{A}}$.

The game of $\mathcal{B}$ attacking Σ_{kpabe} is played in the bilinear group $\mathbb{G}$ of order $N = p_1p_2p_3$. Let $\mathbb{G}_{p_1}$ and $\mathbb{G}_{p_3}$ be the subgroups of order p_1 and p_3 in $\mathbb{G}$ respectively.

Setup. $\mathcal{B}$ receives the public parameter $\mathsf{PP}^{\mathsf{kpabe}} = (N, g, E = e(g,g)^{\alpha}, \ \{U_x = g^{a_x}\}_{x\in\mathcal{U}})$ from the challenger, where $g \in \mathbb{G}_{p_1}$ is a generator of $\mathbb{G}_{p_1}$, and $\alpha, a_x (x \in \mathcal{U}) \in \mathbb{Z}_N$ are random exponents. $\mathcal{B}$ randomly chooses $\eta \in \mathbb{Z}_N, \ \{\alpha_i, \ r_i, \ z_i \in \mathbb{Z}_N\}_{i\in[m]}, \ \{c_j \in \mathbb{Z}_N\}_{j\in[m]}$. Then $\mathcal{B}$ gives $\mathcal{A}$ the following public parameter PP:

$$g, \ f = g^{\eta}, \ E, \ \{E_i = e(g,g)^{\alpha_i}, \ G_i = g^{r_i}, \ Z_i = g^{z_i}\}_{i\in[m]}, \ \{H_j = g^{c_j}\}_{j\in[m]}, \ \ \{U_x\}_{x\in\mathcal{U}}.$$

Phase 1. $\mathcal{A}$ issues adaptive private key queries. To respond to a query for $((i,j),(A,\rho))$, $\mathcal{B}$ submits (A,ρ) to the challenger, and receives a decryption key

$$\mathsf{SK}^{\mathsf{kpabe}}_{(A,\rho)} = \big((A,\rho), \ \{\tilde{K}_{k,1} = g^{(A_k\cdot\boldsymbol{w})}U^{\xi_k}_{\rho(k)}R_{k,1}, \ \ \tilde{K}_{k,2} = g^{\xi_k}R_{k,2}\}^{l}_{k=1}\big),$$

where $\boldsymbol{w} = (\alpha, w_2, \ldots, w_n) \in \mathbb{Z}^n_N$ and $\{\xi_k \in \mathbb{Z}_N, \ R_{k,1}, R_{k,2} \in \mathbb{G}_{p_3}\}^{l}_{k=1}$ are randomly chosen and unknown to $\mathcal{B}$.
$\mathcal{B}$ randomly chooses $\boldsymbol{u} = (\sigma_{i,j}, u_2, \cdots u_n) \in \mathbb{Z}^n_N$, then gives $\mathcal{A}$ a private key $\mathsf{SK}_{(i,j),(A,\rho)} = \big((i,j),(A,\rho), \ K_{i,j}, \ K'_{i,j}, \ K''_{i,j}, \ \{K_{i,j,k,1}, K_{i,j,k,2}\}^{l}_{k=1}\big)$ where

$$K_{i,j} = g^{\alpha_i}g^{r_ic_j}f^{\sigma_{i,j}}, \ K'_{i,j} = g^{\sigma_{i,j}}, \ K''_{i,j} = Z_i^{\sigma_{i,j}}, \ \ \{K_{i,j,k,1} = f^{(A_k\cdot\boldsymbol{u})}\tilde{K}_{k,1}, \ K_{i,j,k,2} = \tilde{K}_{k,2}\}^{l}_{k=1}.$$

Challenge. $\mathcal{A}$ submits to $\mathcal{B}$ an attribute set S^* and two equal-length messages M_0, M_1. $\mathcal{B}$ submits (S^*, M_0, M_1) to the challenger. Note that S^* satisfies the restriction on $\mathcal{B}$ in the KP-ABE security game, i.e. none of the access structures that $\mathcal{B}$ submitted to the challenger in Phase 1 is satisfied by S^*. $\mathcal{B}$ receives the challenge ciphertext in the form of

$$CT^{\mathsf{kpabe}} = \langle S^*, \ \ \tilde{C} = M_b \cdot E^{\pi}, \ \tilde{C}_0 = g^{\pi}, \ \ \{\tilde{C}_x = U_x^{\pi}\}_{x\in S^*}\rangle,$$

where $\pi \in \mathbb{Z}_N$ is randomly chosen and unknown to $\mathcal{B}$.
$\mathcal{B}$ randomly chooses $\kappa, \tau, s_1, \ldots, s_m, t_1, \ldots, t_m \ \in \mathbb{Z}_N, \ \ \boldsymbol{v}_c, \ \boldsymbol{w}_1, \ldots, \boldsymbol{w}_m \ \in \mathbb{Z}^3_N$. In addition, $\mathcal{B}$ randomly chooses $r_x, r_y, r_z \in \mathbb{Z}_N$, and sets $\boldsymbol{\chi}_1 = (r_x, 0, r_z)$, $\boldsymbol{\chi}_2 = (0, r_y, r_z)$, $\boldsymbol{\chi}_3 = \boldsymbol{\chi}_1 \times \boldsymbol{\chi}_2 = (-r_yr_z, -r_xr_z, r_xr_y)$. Then it randomly chooses $\boldsymbol{v}_1 \in \mathbb{Z}^3_N, \ \boldsymbol{v}_i \in span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\} \ \forall i \in \{2, \ldots, m\}$, and creates the ciphertext $\langle S, \ (\boldsymbol{R}_i, \boldsymbol{R}'_i, Q_i, Q'_i, Q''_i, T_i)^{m}_{i=1}, \ (\boldsymbol{C}_j, \boldsymbol{C}'_j)^{m}_{j=1}, \ P, \{P_x\}_{x\in S}\rangle$ for $(\bar{i} = 1, \bar{j} = 1)$ as follows:

1. For each row $i \in [m]$: since $\bar{i} = 1$, it sets

$$\boldsymbol{R}_i = G_i^{s_i\boldsymbol{v}_i}, \boldsymbol{R}'_i = G_i^{\kappa s_i\boldsymbol{v}_i}, \ Q_i = g^{\tau s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_c)}, Q'_i = f^{\tau s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_c)}Z_i^{t_i}\tilde{C}_0^{\eta}, Q''_i = g^{t_i}, \ T_i = \tilde{C}\cdot E_i^{\tau s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_c)}.$$

2. For each column $j \in [m]$: since $\bar{j} = 1$, it sets $\boldsymbol{C}_j = H_j^{\tau \boldsymbol{v}_c} \cdot g^{\kappa \boldsymbol{w}_j}$, $\boldsymbol{C}'_j = g^{\boldsymbol{w}_j}$.
3. It sets $P = \tilde{C}_0$, $\{P_x = \tilde{C}_x\}_{x \in S^*}$.

Phase 2. Same with Phase 1.
Guess. $\mathcal{A}$ gives $\mathcal{B}$ a b'. $\mathcal{B}$ gives b' to the challenger.

Note that the distributions of the public parameter, private keys and challenge ciphertext that $\mathcal{B}$ gives $\mathcal{A}$ are same as the real scheme, we have $Adv_{\mathcal{B}}\Sigma_{\mathsf{kpabe}} = \mathsf{MH}_1^{\mathsf{A}}\mathsf{Adv}_{\mathcal{A}}$.

B.2 Proof of Lemma 1

Proof. Suppose there exists a PPT adversary $\mathcal{A}$ that can selectively break the index-hiding game with advantage ϵ. We build a PPT algorithm $\mathcal{B}$ to solve a Decision 3-Party Diffie-Hellman problem instance as follows.

$\mathcal{B}$ receives a Decision 3-Party Diffie-Hellman problem instance from the challenger as $(g, A = g^a, B = g^b, C = g^c, T)$. The problem instance will be given in the subgroup $\mathbb{G}_{p_1}$ of prime order p_1 in a composite order group $\mathbb{G}$ of order $N = p_1p_2p_3$, i.e., $g \in \mathbb{G}_{p_1}$, $a, b, c \in \mathbb{Z}_{p_1}$, $\mathcal{B}$ is given the factors p_1, p_2, p_3, and its goal is to determine whether $T = g^{abc}$ or a random element from $\mathbb{G}_{p_1}$[8].

Init. $\mathcal{A}$ gives $\mathcal{B}$ the challenge attribute set $S^* \subseteq \mathcal{U}$.
Setup. $\mathcal{B}$ chooses random exponents

$$\eta, \alpha \in \mathbb{Z}_N,\ \{\alpha_i \in \mathbb{Z}_N\}_{i\in[m]},\ \{r_i,\ z'_i \in \mathbb{Z}_N\}_{i\in[m]\setminus\{\bar{i}\}},\ \{c_j \in \mathbb{Z}_N\}_{j\in[m]\setminus\{\bar{j}\}},\quad r'_{\bar{i}},\ z_{\bar{i}},\ c'_{\bar{j}} \in \mathbb{Z}_N,$$
$$\{a_x \in \mathbb{Z}_N\}_{x\in S^*},\quad \{a'_x \in \mathbb{Z}_N\}_{x\in\mathcal{U}\setminus S^*}.$$

$\mathcal{B}$ gives $\mathcal{A}$ the following public parameter PP:

$$g,\ f = C^{\eta},\ E = e(g,g)^{\alpha},\quad \{E_i = e(g,g)^{\alpha_i}\}_{i\in[m]},$$
$$\{G_i = g^{r_i},\ Z_i = C^{z'_i}\}_{i\in[m]\setminus\{\bar{i}\}},\ \{H_j = g^{c_j}\}_{j\in[m]\setminus\{\bar{j}\}},\quad G_{\bar{i}} = B^{r'_{\bar{i}}},\ Z_{\bar{i}} = g^{z_{\bar{i}}},\ H_{\bar{j}} = C^{c'_{\bar{j}}},$$
$$\{U_x = g^{a_x}\}_{x\in S^*},\quad \{U_x = C^{a'_x}\}_{x\in\mathcal{U}\setminus S^*}.$$

Note that $\mathcal{B}$ implicitly chooses $r_{\bar{i}},\ z_i (i \in [m] \setminus \{\bar{i}\}),\ c_{\bar{j}},\ a_x (x \in \mathcal{U} \setminus S^*)\ \in \mathbb{Z}_N$ such that

$$br'_{\bar{i}} \equiv r_{\bar{i}} \bmod p_1,\ cz'_i \equiv z_i \bmod p_1\ \forall i \in [m] \setminus \{\bar{i}\},\ cc'_{\bar{j}} \equiv c_{\bar{j}} \bmod p_1,\ ca'_x \equiv a_x \bmod p_1\ \forall x \in \mathcal{U} \setminus S^*.$$

Key Query. To respond to a query from $\mathcal{A}$ for $((i,j),(A,\rho))$ where A is an $l \times n$ matrix:
– If $(i,j) \neq (\bar{i},\bar{j})$: $\mathcal{B}$ randomly chooses $\boldsymbol{u} = (\sigma_{i,j}, u_2, \ldots, u_n) \in \mathbb{Z}_N^n$, $w_2, \ldots, w_n \in \mathbb{Z}_N$ and $\{\xi_k \in \mathbb{Z}_N, R_{k,1}, R_{k,2} \in \mathbb{G}_{p_3}\}_{k=1}^{l}$. Let $\boldsymbol{w} = (\alpha, w_2, \ldots, w_n)$, $\mathcal{B}$ creates the private key $\mathsf{SK}_{(i,j),(A,\rho)} = \big((i,j),(A,\rho),\ K_{i,j}, K'_{i,j}, K''_{i,j}, \{K_{i,j,k,1}, K_{i,j,k,2}\}_{k=1}^{l}\big)$ as

$$K_{i,j} = \begin{cases} g^{\alpha_i} g^{r_i c_j} f^{\sigma_{i,j}}, & : i \neq \bar{i}, j \neq \bar{j} \\ g^{\alpha_i} B^{r'_{\bar{i}} c_j} f^{\sigma_{i,j}}, & : i = \bar{i}, j \neq \bar{j} \\ g^{\alpha_i} C^{r_i c'_{\bar{j}}} f^{\sigma_{i,j}}, & : i \neq \bar{i}, j = \bar{j}. \end{cases}$$
$$K'_{i,j} = g^{\sigma_{i,j}},\quad K''_{i,j} = Z_i^{\sigma_{i,j}},$$
$$\{K_{i,j,k,1} = f^{(A_k \cdot \boldsymbol{u})} g^{(A_k \cdot \boldsymbol{w})} U_{\rho(k)}^{\xi_k} R_{k,1},\quad K_{i,j,k,2} = g^{\xi_k} R_{k,2}\}_{k=1}^{l}.$$

[8] The situation is similar to that of the proof in [4,5] in the sense that the challenge is given in a subgroup of a composite order group and the factors are given to the simulator. Actually, Lewko and Waters [16] use this case explicitly as an assumption, i.e. the 3-Party Diffie-Hellman Assumption in a Subgroup.

– If $(i,j)=(\bar{i},\bar{j})$: $\mathcal{B}$ randomly chooses $\sigma'_{\bar{i},\bar{j}}, u'_2,\ldots,u'_n, w_2,\ldots,w_n \in \mathbb{Z}_N$, $\{\xi_k \in \mathbb{Z}_N\}_{k\in[l]\ s.t.\ \rho(k)\in S^*}$, $\{\xi'_k \in \mathbb{Z}_N\}_{k\in[l]\ s.t.\ \rho(k)\notin S^*}$, $\{R_{k,1}, R_{k,2} \in \mathbb{G}_{p_3}\}_{k=1}^{l}$. Let $\boldsymbol{u}'=(0,u'_2,\ldots,u'_n)$, $\boldsymbol{w}=(\alpha,w_2,\ldots,w_n) \in \mathbb{Z}_N^n$. As (A,ρ) cannot be satisfied by S^* (since $(i,j)=(\bar{i},\bar{j})$), $\mathcal{B}$ can efficiently find a vector $\boldsymbol{u}''=(u''_1,u''_2,\ldots,u''_n) \in \mathbb{Z}_N^n$ such that $u''_1=1$ and $A_k\cdot\boldsymbol{u}''=0$ for all k such that $\rho(k)\in S^*$. Then implicitly setting $\sigma_{\bar{i},\bar{j}} \in \mathbb{Z}_N$, $\boldsymbol{u}\in\mathbb{Z}_N^n$, $\{\xi_k \in \mathbb{Z}_N\}_{k\in[l]\ s.t.\ \rho(k)\notin S^*}$ as

$$\sigma'_{\bar{i},\bar{j}} - br'_{\bar{i}}c'_{\bar{j}}/\eta \equiv \sigma_{\bar{i},\bar{j}} \bmod p_1,\quad \boldsymbol{u}=\boldsymbol{u}'+\sigma_{\bar{i},\bar{j}}\boldsymbol{u}'',$$
$$\xi'_k + br'_{\bar{i}}c'_{\bar{j}}(A_k\cdot\boldsymbol{u}'')/a'_{\rho(k)} \equiv \xi_k \bmod p_1\ \forall k\in[l]\ s.t.\ \rho(k)\notin S^*,$$

$\mathcal{B}$ creates the private key $\mathsf{SK}_{(\bar{i},\bar{j}),(A,\rho)} = \big((\bar{i},\bar{j}),(A,\rho),\ K_{\bar{i},\bar{j}}, K'_{\bar{i},\bar{j}}, K''_{\bar{i},\bar{j}}, \{K_{\bar{i},\bar{j},k,1}, K_{\bar{i},\bar{j},k,2}\}_{k=1}^{l}\big)$ as:

$$K_{\bar{i},\bar{j}} = g^{\alpha_{\bar{i}}} f^{\sigma'_{\bar{i},\bar{j}}},\quad K'_{\bar{i},\bar{j}} = g^{\sigma'_{\bar{i},\bar{j}}} B^{-r'_{\bar{i}}c'_{\bar{j}}/\eta},\ K''_{\bar{i},\bar{j}} = (g^{\sigma'_{\bar{i},\bar{j}}} B^{-r'_{\bar{i}}c'_{\bar{j}}/\eta})^{z_{\bar{i}}},$$
$$\{K_{i,j,k,1} = f^{(A_k\cdot\boldsymbol{u}')} g^{(A_k\cdot\boldsymbol{w})} U_{\rho(k)}^{\xi_k} R_{k,1},\ K_{i,j,k,2} = g^{\xi_k} R_{k,2}\}_{\rho(k)\in S^*},$$
$$\{K_{i,j,k,1} = f^{(A_k\cdot\boldsymbol{u}')+\sigma'_{\bar{i},\bar{j}}(A_k\cdot\boldsymbol{u}'')} g^{(A_k\cdot\boldsymbol{w})} U_{\rho(k)}^{\xi'_k} R_{k,1},\ K_{i,j,k,2} = g^{\xi'_k} B^{r'_{\bar{i}}c'_{\bar{j}}(A_k\cdot\boldsymbol{u}'')/a'_{\rho(k)}} R_{k,2}\}_{\rho(k)\notin S^*}.$$

Challenge. $\mathcal{A}$ submits a message M. $\mathcal{B}$ randomly chooses

$$\pi',\ \tau',\ s_1,\ldots,s_{\bar{i}-1},s'_{\bar{i}},s_{\bar{i}+1},\ldots,s_m,\ t'_1,\ldots,t'_{\bar{i}-1},t_{\bar{i}},t'_{\bar{i}+1},\ldots,t'_m \in \mathbb{Z}_N,$$
$$\boldsymbol{w}_1,\ldots,\boldsymbol{w}_{\bar{j}-1},\boldsymbol{w}'_{\bar{j}},\ldots,\boldsymbol{w}'_m \in \mathbb{Z}_N^3.$$

$\mathcal{B}$ randomly chooses $r_x,r_y,r_z\in\mathbb{Z}_N$, and sets $\boldsymbol{\chi}_1=(r_x,0,r_z), \boldsymbol{\chi}_2=(0,r_y,r_z), \boldsymbol{\chi}_3=\boldsymbol{\chi}_1\times\boldsymbol{\chi}_2 = (-r_yr_z,-r_xr_z,r_xr_y)$. Then $\mathcal{B}$ randomly chooses

$$\boldsymbol{v}_i \in \mathbb{Z}_N^3\ \forall i\in\{1,\ldots,\bar{i}\},$$
$$\boldsymbol{v}_i \in span\{\boldsymbol{\chi}_1,\boldsymbol{\chi}_2\}\ \forall i\in\{\bar{i}+1,\ldots,m\},$$
$$\boldsymbol{v}_{c,p}\in span\{\boldsymbol{\chi}_1,\boldsymbol{\chi}_2\},\ \boldsymbol{v}_{c,q}=\nu_3\boldsymbol{\chi}_3\in span\{\boldsymbol{\chi}_3\}.$$

$\mathcal{B}$ sets the value of $\pi,\kappa,\tau,s_{\bar{i}},t_i(i\in[m]\backslash\{\bar{i}\})\in\mathbb{Z}_N$, $\boldsymbol{v}_c\in\mathbb{Z}_N^3$, $\{\boldsymbol{w}_j\in\mathbb{Z}_N^3\}_{j=\bar{j}}^m$ by implicitly setting

$$\pi' - a\tau' s'_{\bar{i}}(\boldsymbol{v}_{\bar{i}}\cdot\boldsymbol{v}_{c,q}) \equiv \pi \bmod p_1,\quad b\equiv\kappa \bmod p_1,\quad ab\tau'\equiv\tau \bmod p_1,\quad s'_{\bar{i}}/b \equiv s_{\bar{i}} \bmod p_1,$$
$$t'_i + \eta a\tau' s'_{\bar{i}}(\boldsymbol{v}_{\bar{i}}\cdot\boldsymbol{v}_{c,q})/z'_i \equiv t_i \bmod p_1\ \forall i\in\{1,\ldots,\bar{i}-1\},$$
$$t'_i - \eta b\tau' s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_{c,p})/z'_i + \eta a\tau' s'_{\bar{i}}(\boldsymbol{v}_{\bar{i}}\cdot\boldsymbol{v}_{c,q})/z'_i \equiv t_i \bmod p_1\ \forall i\in\{\bar{i}+1,\ldots,m\},$$
$$\boldsymbol{v}_c = a^{-1}\boldsymbol{v}_{c,p}+\boldsymbol{v}_{c,q},$$
$$\boldsymbol{w}'_{\bar{j}} - cc'_{\bar{j}}\tau'\boldsymbol{v}_{c,p} \equiv \boldsymbol{w}_{\bar{j}} \bmod p_1,$$
$$\boldsymbol{w}'_j - ac_j\tau'\boldsymbol{v}_{c,q} \equiv \boldsymbol{w}_j \bmod p_1\ \forall j\in\{\bar{j}+1,\ldots,m\}.$$

$\mathcal{B}$ creates the ciphertext $\langle S^*,\ (\boldsymbol{R}_i,\boldsymbol{R}'_i,Q_i,Q'_i,Q''_i,T_i)_{i=1}^m,\ (\boldsymbol{C}_j,\boldsymbol{C}'_j)_{j=1}^m,\ P,\{P_x\}_{x\in S^*}\rangle$ as follows:

1. For each $i\in[m]$:
   – if $i<\bar{i}$: it randomly chooses $\hat{s}_i\in\mathbb{Z}_N$, then sets

$$\boldsymbol{R}_i = g^{\boldsymbol{v}_i},\ \boldsymbol{R}'_i = B^{\boldsymbol{v}_i},\quad Q_i = g^{s_i},\ Q'_i = f^{s_i}Z_i^{t'_i}f^{\pi'},\ Q''_i = g^{t'_i}A^{\eta\tau' s'_{\bar{i}}(\boldsymbol{v}_{\bar{i}}\cdot\boldsymbol{v}_{c,q})/z'_i},\quad T_i = E_i^{\hat{s}_i}.$$

– if $i = \bar{i}$: it sets

$$\boldsymbol{R_i} = g^{r'_i s'_{\bar{i}} \boldsymbol{v}_{\bar{i}}}, \quad \boldsymbol{R'_i} = B^{r'_{\bar{i}} s'_i \boldsymbol{v}_{\bar{i}}},$$
$$Q_i = g^{\tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,p})} A^{\tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,q})}, \; Q'_i = C^{\eta \tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,p})} Z_i^{t_{\bar{i}}} f^{\pi'}, \; Q''_i = g^{t_{\bar{i}}},$$
$$T_i = M \cdot e(g^{\alpha_i}, Q_i) \cdot e(g^{\alpha}, g)^{\pi'} \cdot e(g^{\alpha}, A)^{-\tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,q})}.$$

– if $i > \bar{i}$: it sets

$$\boldsymbol{R_i} = g^{r_i s_i \boldsymbol{v}_i}, \quad \boldsymbol{R'_i} = B^{r_i s_i \boldsymbol{v}_i},$$
$$Q_i = B^{\tau' s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_{c,p})}, \; Q'_i = Z_i^{t'_i} f^{\pi'}, \; Q''_i = g^{t'_i} B^{-\eta \tau' s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_{c,p})/z'_i} A^{\eta \tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,q})/z'_i},$$
$$T_i = M \cdot e(g^{\alpha_i}, Q_i) \cdot e(g^{\alpha}, g)^{\pi'} \cdot e(g^{\alpha}, A)^{-\tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,q})}.$$

2. For each $j \in [m]$:
   – if $j < \bar{j}$: it randomly chooses $\mu'_j \in \mathbb{Z}_N$ and implicitly sets the value of μ_j such that $(ab)^{-1}\mu'_j \nu_3 - \nu_3 \equiv \mu_j \bmod N$, then sets $\boldsymbol{C}_j = B^{c_j \tau' \boldsymbol{v}_{c,p}} \cdot g^{c_j \tau' \mu'_j \boldsymbol{v}_{c,q}} \cdot B^{\boldsymbol{w}_j}, \quad \boldsymbol{C'}_j = g^{\boldsymbol{w}_j}$.
   – if $j = \bar{j}$: it sets $\boldsymbol{C}_j = T^{c'_{\bar{j}} \tau' \boldsymbol{v}_{c,q}} \cdot B^{\boldsymbol{w}'_j}, \quad \boldsymbol{C'}_j = g^{\boldsymbol{w}'_{\bar{j}}} \cdot C^{-c'_{\bar{j}} \tau' \boldsymbol{v}_{c,p}}$.
   – if $j > \bar{j}$: it sets $\boldsymbol{C}_j = B^{c_j \tau' \boldsymbol{v}_{c,p}} \cdot B^{\boldsymbol{w}'_j}, \quad \boldsymbol{C'}_j = g^{\boldsymbol{w}'_j} \cdot A^{-c_j \tau' \boldsymbol{v}_{c,q}}$.
3. $P = g^{\pi'} A^{-\tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,q})}, \quad P_x = (g^{\pi'} A^{-\tau' s'_{\bar{i}} (\boldsymbol{v}_{\bar{i}} \cdot \boldsymbol{v}_{c,q})})^{a_x} \; \forall x \in S^*$.

If $T = g^{abc}$, then the ciphertext is a well-formed encryption to the index $(\bar{i}, \bar{j})$. If T is randomly chosen, say $T = g^r$ for some random $r \in \mathbb{Z}_{p_1}$, the ciphertext is a well-formed encryption to the index $(\bar{i}, \bar{j} + 1)$ with implicitly setting $\mu_{\bar{j}}$ such that $(\frac{r}{abc} - 1)\nu_3 \equiv \mu_{\bar{j}} \bmod p_1$.

Guess. $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$ to $\mathcal{B}$, then $\mathcal{B}$ outputs this b' to the challenger as its answer to the Decision 3-Party Diffie-Hellman game.

Note that the distributions of the public parameter, private keys and challenge ciphertext are same as the real scheme, $\mathcal{B}$'s advantage in the Decision 3-Party Diffie-Hellman game will be exactly equal to $\mathcal{A}$'s advantage in selectively breaking the index-hiding game.

B.3 Proof of Claim 2

Proof. Garg et al. [8, Sec. 5.1] proposed an AugBE scheme $\Sigma_{\mathsf{AugBE}} = (\mathsf{Setup}_{\mathsf{AugBE}}, \mathsf{Encrypt}_{\mathsf{AugBE}}, \mathsf{Decrypt}_{\mathsf{AugBE}})$ and proved that it is index-hiding. In their proof of Lemma 6.3 in [8], two hybrid experiments

– H_2^{AugBE}: Encrypt to $(\bar{i}, m + 1)$, (corresponding to H_2 in [8])
– H_3^{AugBE}: Encrypt to $(\bar{i} + 1, 1)$, (corresponding to H_5 in [8])

were defined and proven indistinguishable by a sequence of hybrid sub-experiments. It follows the Lemma 6.3 in [8] that *if the Decisional Linear Assumption and the Decision 3-Party Diffie-Hellman Assumption hold, then for scheme Σ_{AugBE} in [8, Sec. 5.1] no PPT adversary can distinguish between experiments H_2^{AugBE} and H_3^{AugBE} with non-negligible advantage.* To prove our Claim 2, we do not build a direct reduction to the underlying assumptions, instead, we build a reduction to distinguish between experiments H_2^{AugBE} and H_3^{AugBE}. In particular, suppose there is a PPT adversary $\mathcal{A}$ that can distinguish between H_2 and H_3 for our AugKP-ABE scheme Σ_{A} with non-negligible advantage,

we construct a PPT algorithm $\mathcal{B}$ to distinguish between H_2^{AugBE} and H_3^{AugBE} for Σ_{AugBE} with non-negligible advantage.

The game of $\mathcal{B}$ distinguishing between H_2^{AugBE} and H_3^{AugBE} is played in the subgroup $\mathbb{G}_{p_1}$ of order p_1 in a composite order group $\mathbb{G}$ of order $N = p_1p_2p_3$. $\mathcal{B}$ is given the values of p_1, p_2 and p_3. Since the game is played in the subgroup $\mathbb{G}_{p_1}$, $\mathcal{B}$ chooses for itself everything in the subgroup $\mathbb{G}_{p_3}$.

Setup. The challenger gives $\mathcal{B}$ the public key $\mathsf{PK}^{\mathsf{AugBE}}$, and due to $(\bar{i}, m+1) \notin \{(i,j)|1 \le i,j \le m\}$, the challenger gives $\mathcal{B}$ all private keys in the set $\{\mathsf{SK}_{(i,j)}^{\mathsf{AugBE}}|1 \le i,j \le m\}$ as follows:[9]

$$\mathsf{PK}^{\mathsf{AugBE}} = \big(\ g,\ \{E_i = e(g,g)^{\alpha_i},\ G_i = g^{r_i}\}_{i\in[m]},\ \{H_j = g^{c_j},\ f_j\}_{j\in[m]}\ \big),$$
$$\mathsf{SK}_{(i,j)}^{\mathsf{AugBE}} = \big(\tilde{K}_{i,j},\ \tilde{K}'_{i,j},\ \ \{\tilde{K}_{i,j,\tilde{j}}\}_{\tilde{j}\in[m]\setminus\{j\}}\ \big) = \big(g^{\alpha_i} g^{r_i c_j} f_j^{\sigma_{i,j}},\ g^{\sigma_{i,j}},\ \ \{f_{\tilde{j}}^{\sigma_{i,j}}\}_{\tilde{j}\in[m]\setminus\{j\}}\ \big),$$

where $g, f_1, \ldots, f_m \in \mathbb{G}_{p_1}$ and $\{r_i, \alpha_i \in \mathbb{Z}_{p_1}\}_{i\in[m]},\ \{c_j \in \mathbb{Z}_{p_1}\}_{j\in[m]}$, $\sigma_{i,j} (1 \le i,j \le m) \in \mathbb{Z}_{p_1}$ are randomly chosen.
$\mathcal{B}$ randomly chooses $\alpha,\ z_1, \ldots, z_m,\ a'_x (x \in \mathcal{U}) \in \mathbb{Z}_N$, and gives $\mathcal{A}$ the public parameter PP:

$$g,\ f = \prod_{j\in[m]} f_j,\ E = e(g,g)^{\alpha},\ \ \{E_i,\ G_i,\ Z_i = g^{z_i}\}_{i\in[m]},\ \ \{H_j\}_{j\in[m]},\ \ \{U_x = f g^{a'_x}\}_{x\in\mathcal{U}}.$$

Note that $\mathcal{B}$ implicitly chooses $\{a_x \in \mathbb{Z}_N\}_{x\in\mathcal{U}}$ such that $\eta + a'_x \equiv a_x \bmod p_1$ where η satisfies $f = g^{\eta}$.

Key Query. $\mathcal{A}$ issues adaptive private key queries. To respond to a query for $((i,j),(A,\rho))$, where A is an $l \times n$ matrix, $\mathcal{B}$ randomly chooses $u_2, \ldots, u_n,\ w_2, \ldots, w_n \in \mathbb{Z}_N$, and $\{\xi'_k \in \mathbb{Z}_N, R_{k,1}, R_{k,2} \in \mathbb{G}_{p_3}\}_{k=1}^{l}$. For $k = 1$ to l, let $A_k = (A_{k,1}, \ldots, A_{k,n}) \in \mathbb{Z}_N^n$ be the k^{th} row of A. Let $\boldsymbol{w} = (\alpha, w_2, \ldots, w_n)$. $\mathcal{B}$ sets the value of $\boldsymbol{u} \in \mathbb{Z}_N^n, \{\xi_k \in \mathbb{Z}_N\}_{k\in[l]}$ by implicitly setting

$$\boldsymbol{u} = (\sigma_{i,j}, u_2, \ldots, u_n), \quad \xi'_k - \sigma_{i,j} A_{k,1} \equiv \xi_k \bmod p_1\ \forall k \in [l].$$

$\mathcal{B}$ creates a private key $\mathsf{SK}_{(i,j),(A,\rho)} = \big((i,j),(A,\rho),\ K_{i,j}, K'_{i,j}, K''_{i,j},\ \{K_{i,j,k,1}, K_{i,j,k,2}\}_{k=1}^{l}\big)$ from $\mathsf{SK}_{(i,j)}^{\mathsf{AugBE}}$ as follows:

$$K_{i,j} = \tilde{K}_{i,j} \cdot \prod_{\tilde{j}\in[m]\setminus\{j\}} \tilde{K}_{i,j,\tilde{j}},\quad K'_{i,j} = \tilde{K}'_{i,j},\quad K''_{i,j} = (\tilde{K}'_{i,j})^{z_i},$$
$$\{K_{i,j,k,1} = f^{\sum_{d=2}^{n} u_d A_{k,d}} g^{(A_k\cdot\boldsymbol{w})} U_{\rho(k)}^{\xi'_k} (\tilde{K}'_{i,j})^{-a'_{\rho(k)} A_{k,1}} R_{k,1},\ K_{i,j,k,2} = g^{\xi'_k} (\tilde{K}'_{i,j})^{-A_{k,1}} R_{k,2}\}_{k=1}^{l}.$$

Challenge. $\mathcal{A}$ submits a message M and an attribute set S^*. $\mathcal{B}$ sets $Y = \{(i,j)|1 \le i,j \le m\}$ and submits (M, Y) to the challenger. Note that Y satisfies the restriction on $\mathcal{B}$ in the index-hiding game for Σ_{AugBE}, since $(\bar{i}, m+1) \notin Y$. The challenger gives $\mathcal{B}$ the challenge ciphertext $CT^{\mathsf{AugBE}} = \langle(\tilde{\boldsymbol{R}}_i, \tilde{\boldsymbol{R}}'_i, \tilde{Q}_i, \tilde{Q}'_i, \tilde{T}_i)_{i=1}^{m},\ (\tilde{\boldsymbol{C}}_j, \tilde{\boldsymbol{C}}'_j)_{j=1}^{m},\ Y\rangle$, which is encrypted to $(i^*, j^*) \in \{(\bar{i}, m+1), (\bar{i}+1, 1)\}$ and in the form of

1. For each $i \in [m]$:
   - if $i < i^*$: $\tilde{\boldsymbol{R}}_i = g^{\boldsymbol{v}_i},\ \ \tilde{\boldsymbol{R}}'_i = g^{\kappa\boldsymbol{v}_i},\ \ \tilde{Q}_i = g^{s_i},\ \ \tilde{Q}'_i = (\prod_{\hat{j}\in Y_i} f_{\hat{j}})^{s_i},\ \ \tilde{T}_i = E_i^{\hat{s}_i}$.
   - if $i \ge i^*$: $\tilde{\boldsymbol{R}}_i = G_i^{s_i\boldsymbol{v}_i},\ \tilde{\boldsymbol{R}}'_i = G_i^{\kappa s_i\boldsymbol{v}_i},\ \tilde{Q}_i = g^{\tau s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_c)},\ \tilde{Q}'_i = (\prod_{\hat{j}\in Y_i} f_{\hat{j}})^{\tau s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_c)},\ \tilde{T}_i = M \cdot E_i^{\tau s_i(\boldsymbol{v}_i\cdot\boldsymbol{v}_c)}$.

[9] Note that we slightly changed the variable names in the underlying AugBE scheme to better suit our proof.

2. For each $j \in [m]$:
   - if $j < j^*$: $\tilde{\boldsymbol{C}}_j = H_j^{\tau(\boldsymbol{v}_c + \mu_j \boldsymbol{\chi}_3)} \cdot g^{\kappa \boldsymbol{w}_j}$, $\tilde{\boldsymbol{C}}'_j = g^{\boldsymbol{w}_j}$.
   - if $j \geq j^*$: $\tilde{\boldsymbol{C}}_j = H_j^{\tau \boldsymbol{v}_c} \cdot g^{\kappa \boldsymbol{w}_j}$, $\tilde{\boldsymbol{C}}'_j = g^{\boldsymbol{w}_j}$.

Note that $\kappa, \tau, s_i (1 \leq i \leq m), \hat{s}_i (1 \leq i < i^*), \mu_j (1 \leq j < j^*) \in \mathbb{Z}_{p_1}$, $\boldsymbol{v}_c, \boldsymbol{w}_j (1 \leq j \leq m), \boldsymbol{v}_i (1 \leq i \leq i^*) \in \mathbb{Z}_{p_1}^3$, and $\boldsymbol{v}_i (i > i^*) \in span\{\boldsymbol{\chi}_1, \boldsymbol{\chi}_2\}$ are randomly chosen, where $\boldsymbol{\chi}_1 = (r_x, 0, r_z), \boldsymbol{\chi}_2 = (0, r_y, r_z), \boldsymbol{\chi}_3 = \boldsymbol{\chi}_1 \times \boldsymbol{\chi}_2 = (-r_y r_z, -r_x r_z, r_x r_y)$ for randomly chosen $r_x, r_y, r_z \in \mathbb{Z}_{p_1}$, and Y_i denotes the set of all values j such that (i, j) in the set Y, i.e., $Y_i = \{j | (i, j) \in Y\}$.

Note that $Y = \{(i, j) | 1 \leq i, j \leq m\}$ so that $Y_i = \{1, \ldots, m\}$ for all $1 \leq i \leq m$, we have that $\tilde{Q}'_i = (\prod_{\hat{j} \in Y_i} f_{\hat{j}})^{s_i} = f^{s_i}$ for $i < i^*$ and $\tilde{Q}'_i = (\prod_{\hat{j} \in Y_i} f_{\hat{j}})^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)} = f^{\tau s_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_c)}$ for $i \geq i^*$.

$\mathcal{B}$ randomly chooses $\pi, t_1, \ldots, t_m \in \mathbb{Z}_N$, then creates the ciphertext $\langle S^*, (\boldsymbol{R}_i, \boldsymbol{R}'_i, Q_i, Q'_i, Q''_i, T_i)_{i=1}^m, (\boldsymbol{C}_j, \boldsymbol{C}'_j)_{j=1}^m, P, \{P_x\}_{x \in S^*} \rangle$ as follows:

1. For each $i \in [m]$: $\boldsymbol{R}_i = \tilde{\boldsymbol{R}}_i$, $\boldsymbol{R}'_i = \tilde{\boldsymbol{R}}'_i$, $Q_i = \tilde{Q}_i$, $Q'_i = \tilde{Q}'_i \cdot Z_i^{t_i} f^{\pi}$, $Q''_i = g^{t_i}$, $T_i = \tilde{T}_i \cdot E^{\pi}$.
2. For each $j \in [m]$: $\boldsymbol{C}_j = \tilde{\boldsymbol{C}}_j$, $\boldsymbol{C}'_j = \tilde{\boldsymbol{C}}'_j$.
3. $P = g^{\pi}$, $\{P_x = U_x^{\pi}\}_{x \in S^*}$.

Guess. $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$ to $\mathcal{B}$, then $\mathcal{B}$ outputs this b' to the challenger as its answer to distinguish between H_2^{AugBE} and H_3^{AugBE} for scheme Σ_{AugBE}.

As the exponents are applied only to the elements in $\mathbb{G}_{p_1}$, from the view of $\mathcal{A}$, the distributions of the public parameter, private keys and challenge ciphertext that $\mathcal{B}$ gives $\mathcal{A}$ are identical to that in the real scheme. Thus $\mathcal{B}$'s advantage in distinguishing between H_2^{AugBE} and H_3^{AugBE} for scheme Σ_{AugBE} will be exactly equal to $\mathcal{A}$'s advantage in distinguishing between H_2 and H_3 for Σ_{A}.