IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2021CIP0003

Publicized:2021/09/17

This article has been accepted and published on J-STAGE in advance of copyediting. Content is final as presented.

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

Boosting CPA to CCA2 for Leakage-Resilient Attribute-Based Encryption by Using New QA-NIZK

Toi TOMITA[†,††a)], ***Nonmember***, **Wakaha OGATA**[†], ***Member***, ***and*** **Kaoru KUROSAWA**[††,†††], ***Fellow***

SUMMARY In this paper, we construct the first efficient leakage-resilient CCA2 (LR-CCA2)-secure attribute-based encryption (ABE) schemes. We also construct the first efficient LR-CCA2-secure identity-based encryption (IBE) scheme with optimal leakage rate.

To obtain our results, we develop a new quasi-adaptive non-interactive zero-knowledge (QA-NIZK) argument for the ciphertext consistency of the LR-CPA-secure schemes.

Our ABE schemes are obtained by boosting the LR-CPA-security of some existing schemes to the LR-CCA2-security by using our QA-NIZK arguments. The schemes are almost as efficient as the underlying LR-CPA-secure schemes.

key words: *Leakage-resilience, CCA2-security, Attribute-based encryption, QA-NIZK, Simulation-soundness.*

1. Introduction

1.1 Leakage-Resilient Cryptography

Traditional security notions for encryption schemes such as IND-CPA/CCA2 implicitly assume that the secret key is completely hidden from an adversary. However, in the real world, an adversary may learn some partial information on the secret key by side-channel attacks [1] or by cold-boot attacks [2].

To tackle this problem, Akavia et al. [3] introduced the bounded memory leakage (BML) model and formulated leakage-resilient CPA (LR-CPA) security of public-key encryption (PKE) schemes. Soon after, Naor and Segev [4] defined LR-CCA2 security. In the BML model, the total amount of key leakage is bounded. Brakerski et al. [5] and Dodis et al. [6] independently introduced the continual memory leakage (CML) model, where there is a notion of time periods and secret keys are updated at the end of each time period. In the CML model, an adversary is allowed to obtain a limited amount of leakage of secret keys in each time period, but there is no limitation on the total amount of leakage that the adversary obtained in all time periods. An LR-CPA/CCA2-secure PKE scheme in the BML or CML model is IND-CPA/CCA2-secure even if some partial information of the secret key is leaked to the adversary.

We can also consider the leakage-resilient (LR) security model of advanced encryption schemes such as attribute-based encryption (ABE) schemes [3], [7]. Indeed, many efficient LR-CPA-secure ABE schemes have been constructed so far [8]–[11].

To achieve LR-CCA2-security, there exists a generic method to transform any LR-CPA-secure ABE schemes to LR-CCA2-secure ones based on the Naor-Yung double encryption paradigm [12]. The resulting scheme is, however, very inefficient because this method uses a simulation-sound NIZK [4], [13] or a true simulation extractable NIZK [14] in addition to doubling the original (CPA) ciphertext.

Unfortunately, the generic construction is the only known method to construct LR-CCA2-secure ABE scheme except for special cases like identity-based encryption (IBE). A natural open question arises:

Can we construct efficient LR-CCA2-secure ABE schemes?

Next, we focus on IBE. For IBE, Hofheinz et al. [15] presented a (non-black-box) CPA-to-CCA2 transformation by using a quasi-adaptive non-interactive zero-knowledge (QA-NIZK) argument for linear subspaces, introduced by [16]. Their approach is very efficient, but cannot be used in the leakage-resilient setting, as we will explain in Section 1.3.

On the other hand, several LR-CCA2-secure IBE schemes [13], [17]–[21], which are more efficient than the generic construction, have been proposed. However, to the best of our knowledge, no scheme is secure if more than half of the secret key is leaked. Therefore, a second open question that we are interested in is:

Can we construct efficient LR-CCA2-secure IBE schemes that allow leakage of most of the secret key?

1.2 Our Contributions

This paper gives positive answers to the above questions. We develop new LR-CCA2-secure ABE schemes that are more efficient than the generic construction. Our schemes are obtained by boosting the LR-CPA-security of some existing schemes [9], [11] to the LR-CCA2-security. The schemes are almost as efficient as the underlying LR-CPA-secure schemes, and in particular, each ciphertext is only 2 group elements larger than those of the underlying schemes. We summarize our results below.

1. We construct the first LR-CCA2-secure ABE schemes for a large class of predicates. Our ABE scheme allows

[†]Tokyo Institute of Technology, Tokyo, 152-8552, Japan
[††]National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, 135-0064, Japan
[†††]Chuo University, Tokyo, 112-8551, Japan
a) E-mail: tomita.t.ae@m.titech.ac.jp

Copyright © 200x The Institute of Electronics, Information and Communication Engineers

its master secret key leakage and user's secret key in the CML model. By combining with [11], we obtain the following concrete LR-CCA2-secure ABE schemes:

- Inner-product encryption (IPE) and non-zero IPE,
- (Doubly) spatial encryption,
- Key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE) for boolean formulae,
- KP-ABE and CP-ABE for arithmetic formulae,
- Broadcast encryption.

The leakage rates of the above LR-CCA2-secure ABE scheme are the same as the LR-CPA-secure one of [11].

2. We construct the first LR-CCA2-secure IBE scheme with optimal leakage rate[†]. More specifically, our IBE scheme is resilient to the leakage of $(1-o(1))$-fraction of its user's secret key in the BML model, but does not allow its master key leakage.

To obtain our results, we develop a new QA-NIZK argument for the ciphertext consistency of the LR-CPA-secure schemes. Our new QA-NIZK argument has simulation-soundness and a small proof, that allows us to boost LR-CPA-security to LR-CCA2-security efficiently.

A QA-NIZK argument is an NIZK argument in which a common reference string depends on the language. We develop the first simulation-sound QA-NIZK argument for a language that is characterized by *generalized* tagged linear subspaces (GTLS), that is defined as

$$\mathcal{L}_{\rho}^{\mathsf{GTLS}} := \left\{([\mathbf{c}], \mathbf{x}) \mid \exists \mathbf{r} \in \mathbb{Z}_q^t \text{ s.t. } \mathbf{c} = \mathbf{M}_{\mathbf{x}}\mathbf{r}\right\}, \tag{1}$$

where $\rho := ([\mathbf{M}], [\mathbf{M}'_1], \ldots, [\mathbf{M}'_m]) \in \mathbb{G}^{n\times t} \times (\mathbb{G}^{n'\times t})^m$, $n > t$, $n' \geq 1$, x_i is the i-th element of $\mathbf{x} \in \mathbb{Z}_q^m$, and $\mathbf{M}_{\mathbf{x}} := \begin{pmatrix} \mathbf{M} \\ \sum_{i=1}^m x_i \mathbf{M}'_i \end{pmatrix}$. (We use implicit representation of group elements as in [22]. See Section 2.1.) Previously, the simulation-sound QA-NIZK argument is known only for $m = 0$ (linear subspaces). No-simulation-sound QA-NIZK argument is known for $m = 2$ and $x_1 = 1$ (tagged linear subspaces). We also show that a QA-NIZK argument for the above language implies one for the following language:

$$\hat{\mathcal{L}}_{\rho}^{\mathsf{GTLS}} := \left\{([\mathbf{c}], L) \mid \exists \mathbf{r} \in \mathbb{Z}_q^t \text{ s.t. } \mathbf{c} = \mathbf{M}_L\mathbf{r}\right\}, \tag{2}$$

where L is a linear map and $\mathbf{M}_L := \begin{pmatrix} \mathbf{M} \\ L(\mathbf{M}'_1, \ldots, \mathbf{M}'_m) \end{pmatrix}$. We summarize the differences between our QA-NIZK argument and the existing ones in Table 1.

We believe that our new QA-NIZK argument has another applications because it supports more general languages than languages for just linear subspaces before.

1.3 Technical Overview

Here, we provide overviews of our techniques.

[†]The leakage rate is defined as the ratio of the amount of allowed leakage to the secret key size. *Optimal* leakage rate means that the leakage rate can be arbitrarily close to 1 by setting parameters appropriately.

How to boost CPA to CCA2 for leakage-resilient ABE. As mentioned in Section 1.1, we can obtain LR-CCA2-secure schemes from LR-CPA-secure schemes through the Naor-Yung paradigm [12]. The resulting scheme is, however, very inefficient.

In [24]–[26], the authors constructed CCA2-secure PKE schemes by using an efficient simulation-sound QA-NIZK argument for linear subspaces. In these PKE schemes, the ciphertext consistency can be verified by a linear equation, which depends only on a public key. In the case of ABEs, however, the consistency check equation depends not only on the public parameter (which is a fixed parameter) but also on the attribute (which is a variable). Therefore, we cannot use existing (simulation-sound) QA-NIZK arguments for linear subspaces to construct LR-CCA2-secure ABE schemes in general.

On the other hand, in [15], [27], [28], the authors showed a CPA-to-CCA2 transformation for (not leakage-resilient) IBE schemes by using a simulation-sound (tag-based) QA-NIZK argument for linear subspaces. At first glance, thanks to the public verifiability of the QA-NIZK argument, their approach seems to provide LR-CCA2-secure schemes by only replacing the CPA-secure schemes with LR-CPA-secure ones. Unfortunately, it does not work well, because the proof of the (non-LR) CCA2-security in their approach makes use of the property that a secret key is uniformly random from the adversary's view point. In the LR-security model, the adversary can learn partial knowledge about the secret key, and hence we cannot ensure the uniform randomness of it.

From the above discussion, it is difficult to construct LR-CCA2-secure ABE schemes by using the existing very efficient QA-NIZK schemes. We solve this problem by developing a new simulation-sound QA-NIZK argument.

How to achieve simulation-sound QA-NIZK argument for GTLS. Our *simulation sound* QA-NIZK argument for *generalized tagged linear subspaces* is obtained as a (non-trivial) combination of the QA-NIZK arguments by Jutla and Roy [16] and by Kiltz and Wee [23]. The former is *no simulation sound* one for *tagged linear subspaces* and the latter is a *simulation sound* one for *linear subspaces*. Our main observation is to consider designated verifier (DV) variants of these two QA-NIZK arguments. Then their security proofs are greatly simplified, and we find out that these arguments have a close relationship. This observation allows us to construct the first *simulation-sound* QA-NIZK argument for *generalized tagged* linear subspaces.

More details are as follows. The language for tagged linear subspaces is defined as follows:

$$\mathcal{L}_{\rho}^{\mathsf{tagged}} := \left\{([\mathbf{c}], x) \mid \exists \mathbf{r} \in \mathbb{Z}_q^t \text{ s.t. } \mathbf{c} = \mathbf{M}_x\mathbf{r}\right\}, \tag{3}$$

where $\rho := ([\mathbf{M}], [\mathbf{M}'_0], [\mathbf{M}'_1])$ and $\mathbf{M}_x := \begin{pmatrix} \mathbf{M} \\ \mathbf{M}'_0 + x\mathbf{M}'_1 \end{pmatrix}$. In the DV variant of Jutla-Roy's scheme, a verifier has a secret verification key which is random vectors $\mathbf{k} \in \mathbb{Z}_q^{n'}$ and $\mathbf{k}_0, \mathbf{k}_1 \in \mathbb{Z}_q^n$, and the common reference string (CRS) is the projections

Table 1 Summary of the differences between our scheme and existing schemes. The schemes in the column of $m = 0$ are a QA-NIZK argument for just linear subspaces, and the schemes in the column of $m \geq 1$ are a QA-NIZK argument for (generalized) tagged linear subspaces.

	Linear subspaces ($m = 0$)	Tagged linear subspaces ($m \geq 1$)
No-simulation-sound	Already exist, e.g. JR13 [16]	Already exist, e.g. JR13 [16]
Simulation-sound	Already exist, e.g. KW15 [23]	**Ours**

$[\mathbf{p}_0] \coloneqq [\mathbf{M}^\top \mathbf{k}_0 + \mathbf{M}_0'^\top \mathbf{k}]$ and $[\mathbf{p}_1] \coloneqq [\mathbf{M}^\top \mathbf{k}_1 + \mathbf{M}_1'^\top \mathbf{k}]$. To prove that $([\mathbf{c}], x)$ satisfies $[\mathbf{c}] = [\mathbf{M}_x \mathbf{r}]$ for some $\mathbf{r} \in \mathbb{Z}_q^t$, the prover outputs $\pi \coloneqq [\mathbf{r}^\top(\mathbf{p}_0 + x\mathbf{p}_1)]$ as a proof. With the verification key, the (designated) verifier can check whether $\pi = [\mathbf{c}^\top \mathbf{k}_x]$, where $\mathbf{k}_x \coloneqq \binom{\mathbf{k}_0 + x\mathbf{k}_1}{\mathbf{k}}$. By using $\mathbf{k}$, $\mathbf{k}_0$, and $\mathbf{k}_1$ as a simulation trapdoor, a simulated proof is given by $\tilde{\pi} \coloneqq [\mathbf{c}^\top \mathbf{k}_x]$. Perfect completeness and zero-knowledge follow from the following equation:

$$\begin{aligned}\mathbf{r}^\top(\mathbf{p}_0 + x\mathbf{p}_1) &= \mathbf{r}^\top \left(\mathbf{M}^\top(\mathbf{k}_0 + x\mathbf{k}_1) + \left(\mathbf{M}_0'^\top + x\mathbf{M}_1'^\top\right)\mathbf{k}\right) \\ &= \mathbf{r}^\top \mathbf{M}_x^\top \mathbf{k}_x \\ &= \mathbf{c}^\top \mathbf{k}_x.\end{aligned}$$

Soundness is guaranteed by the fact that if $\mathbf{c}$ is outside the span of $\mathbf{M}_x$ for x chosen by an adversary, then $\mathbf{c}^\top \mathbf{k}_x$ is completely random given $\mathbf{M}^\top \mathbf{k}_0 + \mathbf{M}_0'^\top \mathbf{k}$ and $\mathbf{M}^\top \mathbf{k}_1 + \mathbf{M}_1'^\top \mathbf{k}$.

Now we observe that this scheme has similar structure to the DV variant of Kiltz-Wee's scheme. Therefore, by using their techniques, the scheme can be converted to a *simulation-sound* and *publicly-verifiable* QA-NIZK argument.

The above QA-NIZK argument is a special case of a QA-NIZK argument for the *generalized* tagged linear subspaces (GTLS) given by Eq. (1), where $m = 2$ and $x_1 = 1$. This is one of our contribution. We further show that this argument can be extended to the GTLS in this paper.

By a straightforward encoding, we also demonstrate that a QA-NIZK argument for GTLS implies one for the language in Eq. (2). We refer to Section 3.3 for details.

Summary. Finally, we summarize how to construct our LR-CCA2-secure ABE schemes.

1. We start from an LR-CPA-secure ABE scheme that the ciphertext is written as $[\mathbf{c}]$ in Eq. (1), where x is the attribute and $\mathbf{r}$ is a random vector used in encryption. Indeed, the LR-CPA-secure IBE scheme of [9] and the LR-CPA-secure ABE scheme of [11] satisfy this condition.
2. We construct a simulation-sound QA-NIZK argument for the language given by Eq. (1). As mentioned above, previously, the simulation-sound QA-NIZK argument is known *only* for linear subspaces [16] (i.e., $m = 0$ in Eq. (1)), and the *no*-simulation-sound QA-NIZK argument is known even for tagged linear subspaces [23] (i.e., $m = 2$ and $x_1 = 1$ in Eq. (1)). It is not an easy task to extend these results to any m and any $x_1, \ldots, x_m$. Our new QA-NIZK argument is obtained as a (non-trivial) combination of the QA-NIZK arguments by [16] and [23]. Our main observation is to consider designated verifier variants of these two QA-NIZK arguments.
3. The proposed LR-CCA2-secure ABE scheme is obtained by adding the above simulation-sound QA-NIZK proof to the ciphertext $[\mathbf{c}]$.

2. Preliminaries

Notations. We denote $[a, b]$ as the set of $\{a, \ldots, b\}$ for any $a, b \in \mathbb{N}$ with $a \leq b$. We denote the empty string as ϵ and the empty set $\emptyset$. We use $x \leftarrow\!\$\, \mathcal{S}$ to denote the process of sampling an element x from $\mathcal{S}$ uniformly at random if $\mathcal{S}$ is a finite set. We denote the bit length of element x as $|x|$. We denote a security parameter as λ. For integers $k > 1, \eta \in \mathbb{N}$ and a matrix $\mathbf{M} \in \mathbb{Z}_q^{(k+\eta)\times k}$, we denote the upper square matrix of $\mathbf{M}$ as $\overline{\mathbf{M}} \in \mathbb{Z}_q^{k\times k}$ and the lower η rows of $\mathbf{M}$ as $\underline{\mathbf{M}} \in \mathbb{Z}_q^{\eta\times k}$. Similarly, for a column vector $\mathbf{v} \in \mathbb{Z}_q^{k+\eta}$, we denote the upper k elements of $\mathbf{v}$ as $\overline{\mathbf{v}} \in \mathbb{Z}_q^k$ and the lower η elements of $\mathbf{v}$ as $\underline{\mathbf{v}} \in \mathbb{Z}_q^\eta$. $\mathsf{Span}(\mathbf{M}) \coloneqq \left\{\mathbf{M}\mathbf{r} \mid \mathbf{r} \in \mathbb{Z}_q^m\right\} \subset \mathbb{Z}_q^n$ denotes the linear span of $\mathbf{M}$. All the logarithms used in this paper are in base 2.

All algorithms in this paper are probabilistic polynomial time (PPT) unless we state otherwise. If $\mathcal{A}$ is an algorithm, then we write $a \leftarrow\!\$\, \mathcal{A}(b)$ to denote the random variable a outputted by $\mathcal{A}$ on input b.

Games. Following [29], we use code-based games to define and prove security. A game contains procedures INIT and $\mathsf{FINALIZE}$, and some additional procedures $\mathsf{P}_1, \ldots, \mathsf{P}_n$, which are defined in pseudo-code. Initially, all variables and all sets in a game are defined as 0 and empty (i.e., $\emptyset$), respectively. If an adversary $\mathcal{A}$ is executed in game G (denoted by $\mathsf{G}^{\mathcal{A}}$), it first calls INIT and then obtains its output. Next, it may make arbitrary queries to P_i (according to their specification), and obtain their output. Finally, it makes one single call to $\mathsf{FINALIZE}$ and stops, and G outputs d which is the output of $\mathsf{FINALIZE}$. We use $\mathsf{G}^{\mathcal{A}} \Rightarrow d$ to denote that G outputs d after interacting with $\mathcal{A}$.

Collision Resistant Hash Functions. Let $\mathcal{H}$ be a family of hash functions $\mathsf{H} : \{0,1\}^* \to \mathcal{X}$, where $\mathcal{X} = \mathcal{X}_\lambda$ is a finite set. We assume that a hash function H is efficiently samplable from $\mathcal{H}$.

Definition 1 (Collision resistance): We say that a family of hash functions $\mathcal{H}$ is collision-resistant (CR) if for any PPT adversary $\mathcal{A}$,

$$\mathsf{Adv}^{\mathsf{CR}}_{\mathcal{H},\mathcal{A}}(\lambda) \coloneqq \Pr[x \neq x' \wedge \mathsf{H}(x) = \mathsf{H}(x') \mid \mathsf{H} \leftarrow\!\$\, \mathcal{H}, (x, x') \leftarrow\!\$\, \mathcal{A}(1^\lambda, \mathsf{H})]$$

is negligible.

2.1 Pairing Groups and Matrix Diffie-Hellman Assumptions

Let GGen be a PPT algorithm that on input 1^λ returns a description $\mathcal{G} = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, P_1, P_2, e)$ of asymmetric pairing groups, where $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ are cyclic-groups of order q for a λ-bit prime q, P_1 and P_2 are generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficient computable (non-degenerated) bilinear map. Define $P_T := e(P_1, P_2)$, which is a generator in $\mathbb{G}_T$.

We use implicit representation of group elements as in [22]. For $s \in \{1, 2, T\}$ and $a \in \mathbb{Z}_q$, we define $[a]_s := aP_s \in \mathbb{G}_s$ as the implicit representation of a in $\mathbb{G}_s$. Similarly, for $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we define $[\mathbf{A}]_s := \mathbf{A}P_s \in \mathbb{G}_s^{n\times m}$. Note that it is efficient to compute $[\mathbf{AB}]_s =: [\mathbf{A}]_s\mathbf{B} = \mathbf{A}[\mathbf{B}]_s$ given $([\mathbf{A}]_s, \mathbf{B})$ or $(\mathbf{A}, [\mathbf{B}]_s)$ with matching dimensions. We define $[\mathbf{A}]_1 \circ [\mathbf{B}]_2 := e([\mathbf{A}]_1, [\mathbf{B}]_2) = [\mathbf{AB}]_T$, which can be efficiently computed given $[\mathbf{A}]_1$ and $[\mathbf{B}]_2$. From the linearity of $[\cdot]_s$, we have $a[\mathbf{A}]_s + b[\mathbf{B}]_s = [a\mathbf{A} + b\mathbf{B}]_s$ for any $a, b \in \mathbb{Z}_q$ and $[\mathbf{A}]_s, [\mathbf{B}]_s \in \mathbb{G}_s^{n\times m}$.

Throughout this paper, we also use the following useful notations from [30]. Let $L : \mathbb{Z}_q^t \to \mathbb{Z}_q^{t'}$ be a $\mathbb{Z}_q$-linear map. A $\mathbb{Z}_q$-linear map L can be encoded as a matrix $\mathbf{L} = (l_{i,j}) \in \mathbb{Z}_q^{t\times t'}$ such that

$$L : \mathbb{Z}_q^t \ni \begin{pmatrix} w_1 \\ \dots \\ w_t \end{pmatrix} \mapsto \begin{pmatrix} \sum_{i=1}^t l_{i,1} w_i \\ \dots \\ \sum_{i=1}^t l_{i,t'} w_i \end{pmatrix} \in \mathbb{Z}_q^{t'}.$$

We can naturally extend a $\mathbb{Z}_q$-linear map L to a $\mathbb{Z}_q^{n\times m}$-linear map and $\mathbb{G}_s^{n\times m}$-linear map as follows:

$$\begin{pmatrix} \mathbf{W}_1 \\ \vdots \\ \mathbf{W}_t \end{pmatrix} \mapsto \begin{pmatrix} \sum_{i=1}^t l_{i,1}\mathbf{W}_i \\ \vdots \\ \sum_{i=1}^t l_{i,t'}\mathbf{W}_i \end{pmatrix} \text{ and}$$

$$\begin{pmatrix} [\mathbf{W}_1]_s \\ \vdots \\ [\mathbf{W}_t]_s \end{pmatrix} \mapsto \begin{pmatrix} \sum_{i=1}^t l_{i,1}[\mathbf{W}_i]_s \\ \vdots \\ \sum_{i=1}^t l_{i,t'}[\mathbf{W}_i]_s \end{pmatrix},$$

where $\mathbf{W}_1, \dots, \mathbf{W}_t \in \mathbb{Z}_q^{n\times m}$. Because they essentially share the same structure, we employ the same notation L. Here, we highlight the following commutativity:

- $L(\cdot)$ **and** $[\cdot]_s$**:** For any $\mathbf{w} \in \mathbb{Z}_q^t$, we have $L([\mathbf{w}]_s) = [L(\mathbf{w})]_s$.
- $L(\cdot)$ **and multiplication:** For any $\mathbf{W}_1, \dots, \mathbf{W}_t \in \mathbb{Z}_q^{n\times k}$ and $\mathbf{A} \in \mathbb{Z}_q^{k\times m}$, we have $L([\mathbf{W}_1\mathbf{A}]_s, \dots, [\mathbf{W}_t\mathbf{A}]_s) = L([\mathbf{W}_1]_s, \dots, [\mathbf{W}_t]_s)\mathbf{A}$.
- $L(\cdot)$ **and pairing:** For any $\mathbf{A} \in \mathbb{Z}_q^{n\times k}$ and $\mathbf{B}_1, \dots, \mathbf{B}_t \in \mathbb{Z}_q^{k\times m}$, we have $[\mathbf{A}]_1 \circ L([\mathbf{B}_1]_2, \dots, [\mathbf{B}_t]_2) = L([\mathbf{AB}_1]_T, \dots, [\mathbf{AB}_t]_T)$.

In the following, we denote $L(w_1, \dots, w_t)$ as $L\left((w_i)_{i\in[1,t]}\right)$.

Next, we recall the definition of the matrix Diffie-Hellman (MDDH) [22] and related assumptions [31].

Definition 2 (Matrix distribution): Let $k, \ell \in \mathbb{N}$ with $\ell > k$. We call $\mathcal{D}_{\ell,k}$ a *matrix distribution* if it outputs matrices in $\mathbb{Z}_q^{\ell\times k}$ of full rank k in polynomial time. By $\mathcal{D}_k$, we denote $\mathcal{D}_{k+1,k}$.

Without loss of generality, we assume the first k rows of $\mathbf{A} \leftarrow\!\$\, \mathcal{D}_{\ell,k}$ (i.e., $\bar{\mathbf{A}} \in \mathbb{Z}_q^{k\times k}$) form an invertible matrix. For a matrix $\mathbf{A} \leftarrow\!\$\, \mathcal{D}_{\ell,k}$, we define the set of kernel of $\mathbf{A}$ as

$$\ker(\mathbf{A}) := \{\mathbf{A}^\perp \in \mathbb{Z}_q^{\ell\times(\ell-k)} \mid \mathbf{A}^\top\mathbf{A}^\perp = \mathbf{0} \in \mathbb{Z}_q^{k\times(\ell-k)} \text{ and } \mathbf{A}^\perp \text{ has rank } (\ell-k)\}.$$

Given a matrix $\mathbf{A}$ over $\mathbb{Z}_q^{\ell\times k}$, it is efficient to sample an $\mathbf{A}^\perp$ from $\ker(\mathbf{A})$.

The $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman problem is to distinguish the two distributions $([\mathbf{A}]_s, [\mathbf{Aw}]_s)$ and $([\mathbf{A}]_s, [\mathbf{u}]_s)$, where $\mathbf{A} \leftarrow\!\$\, \mathcal{D}_{\ell,k}$, $\mathbf{w} \leftarrow\!\$\, \mathbb{Z}_q^k$ and $\mathbf{u} \leftarrow\!\$\, \mathbb{Z}_q^\ell$.

Definition 3 ($\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman assumption): Let $k \geq 1$ and $\ell > k$ be integers. Let $\mathcal{D}_{\ell,k}$ be a matrix distribution and $s \in \{1, 2, T\}$. We say that the $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman ($\mathcal{D}_{\ell,k}$-MDDH) problem is hard relative to GGen in group $\mathbb{G}_s$ if for any PPT adversary $\mathcal{A}$,

$$\mathsf{Adv}^{\mathsf{MDDH}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{A}}(\lambda) := \big|\Pr[1 \leftarrow\!\$\, \mathcal{A}(\mathcal{G}, [\mathbf{A}]_s, [\mathbf{Aw}]_s)] - \Pr[1 \leftarrow\!\$\, \mathcal{A}(\mathcal{G}, [\mathbf{A}]_s, [\mathbf{u}]_s)]\big|$$

is negligible, where the probability is taken over $\mathcal{G} \leftarrow\!\$\, \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \leftarrow\!\$\, \mathcal{D}_{\ell,k}$, $\mathbf{w} \leftarrow\!\$\, \mathbb{Z}_q^k$, and $\mathbf{u} \leftarrow\!\$\, \mathbb{Z}_q^\ell$.

We define the $\mathcal{D}_{\ell,k}$-Kernel Diffie-Hellman ($\mathcal{D}_{\ell,k}$-KerMDH) assumption [31] which is a natural search variant of the $\mathcal{D}_{\ell,k}$-MDDH assumption.

Definition 4 ($\mathcal{D}_{\ell,k}$-Kernel Diffie-Hellman assumption): Let $k \geq 1$ and $\ell > k$ be integers. Let $\mathcal{D}_{\ell,k}$ be a matrix distribution and $s \in \{1, 2\}$. We say that the $\mathcal{D}_{\ell,k}$-Kernel Diffie-Hellman ($\mathcal{D}_{\ell,k}$-KerMDH) problem is hard relative to GGen in group $\mathbb{G}_s$ if for any PPT adversary $\mathcal{A}$,

$$\mathsf{Adv}^{\mathsf{KerMDH}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{A}}(\lambda) := \Pr[\mathbf{c}^\top\mathbf{A} = \mathbf{0} \wedge \mathbf{c} \neq \mathbf{0} \mid [\mathbf{c}]_{3-s} \leftarrow\!\$\, \mathcal{A}(\mathcal{G}, [\mathbf{A}]_s)]$$

is negligible, where the probability is taken over $\mathcal{G} \leftarrow\!\$\, \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \leftarrow\!\$\, \mathcal{D}_{\ell,k}$.

The following lemma shows that the $\mathcal{D}_{\ell,k}$-KerMDH assumption is a relaxation of the $\mathcal{D}_{\ell,k}$-MDDH assumption.

Lemma 1 ($\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow$ $\mathcal{D}_{\ell,k}$-KerMDH [31]): For any matrix distribution $\mathcal{D}_{\ell,k}$, if $\mathcal{D}_{\ell,k}$-MDDH in group $\mathbb{G}_s$ is hard, then $\mathcal{D}_{\ell,k}$-KerMDH in group $\mathbb{G}_s$ is hard.

We also define the *external* $\mathcal{D}_{\ell,k}$-matrix Diffie-Hellman ($\mathcal{D}_{\ell,k}$-exMDDH) assumption, which is a generalization of the external decision linear assumption [32]. We emphasize that we need $k \geq 2$ to hold this assumption, unlike the above assumptions.

Definition 5 (external $\mathcal{D}_{\ell,k}$-MDDH assumption): Let $k \geq 2$ and $\ell > k$ be integers. Let $\mathcal{D}_{\ell,k}$ be a matrix distribution

and $s \in \{1, 2\}$. We say that the external $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman ($\mathcal{D}_{\ell,k}$-exMDDH) problem is hard relative to GGen in group $\mathbb{G}_s$ if for any PPT adversary $\mathcal{A}$,

$$\mathsf{Adv}^{\mathsf{exMDDH}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{A}}(\lambda) := \big|\Pr[1 \leftarrow\$ \mathcal{A}(\mathcal{G}, [\mathbf{A}]_1, [\mathbf{A}]_2, [\mathbf{Aw}]_s)] - \Pr[1 \leftarrow\$ \mathcal{A}(\mathcal{G}, [\mathbf{A}]_1, [\mathbf{A}]_2, [\mathbf{u}]_s)]\big|$$

is negligible, where the probability is taken over $\mathcal{G} \leftarrow\$ \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \leftarrow\$ \mathcal{D}_{\ell,k}$, $\mathbf{w} \leftarrow\$ \mathbb{Z}_q^k$, and $\mathbf{u} \leftarrow\$ \mathbb{Z}_q^\ell$.

3. Quasi-Adaptive Non-Interactive Zero-Knowledge Argument

A quasi-adaptive non-interactive zero-knowledge (QA-NIZK) argument, introduced by Jutla and Roy [16], is an NIZK argument that the common reference string depends on the language for which proofs are generated.

In this section, we describe a simulation-sound QA-NIZK argument used to boost CPA to CCA2 for leakage-resilient ABE schemes. Our QA-NIZK argument is for *generalized* tagged linear subspaces (GTLS) supporting a language defined as

$$\mathcal{L}_\rho^{\mathsf{GTLS}} := \{([\mathbf{c}]_1, \mathbf{x}) \in \mathbb{G}_1^{n+n'} \times \mathbb{Z}_q^m \mid \exists \mathbf{r} \in \mathbb{Z}_q^t \text{ s.t. } \mathbf{c} = \mathbf{M}_\mathbf{x}\mathbf{r}\},$$

where $\rho := ([\mathbf{M}]_1, [\mathbf{M}'_1]_1, \ldots, [\mathbf{M}'_m]_1) \in \mathbb{G}_1^{n\times t} \times (\mathbb{G}_1^{n'\times t})^m$, x_i is the i-th element of $\mathbf{x}$, and $\mathbf{M}_\mathbf{x} := \begin{pmatrix} \mathbf{M} \\ \sum_{i=1}^m x_i \mathbf{M}'_i \end{pmatrix}$. This is a combination of the QA-NIZK arguments for tagged linear subspaces by Jutla and Roy [16] and a simulation-sound one for linear subspaces by Kiltz and Wee [23].

In Section 3.1, we first give the general definition of QA-NIZK arguments. In Section 3.2, we then describe our one-time simulation-sound QA-NIZK argument for GTLS.[†] In Section 3.3, we show that a QA-NIZK argument for the above language implies one for GTLS expressed by linear maps (Eq. (2)).

3.1 Definition

Let par be a public parameter and $\mathcal{D}_{\mathsf{par}}$ be a probability distribution over a set of strings $\{\rho\}$, that specifies a witness relation R_ρ with a corresponding language $\mathcal{L}_\rho = \{y \mid \exists w \text{ s.t. } R_\rho(y, w) = 1\}$. We recall the formal definition of QA-NIZK arguments for a collection of languages $\mathcal{L} := \{\mathcal{L}_\rho\}_{\rho\in\mathcal{D}_{\mathsf{par}}}$.

Syntax. A QA-NIZK argument for $\mathcal{L}$ consists of the following algorithms $\Pi = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$.

$\mathsf{Gen}(\mathsf{par}, \rho) \to (\mathsf{crs}, \mathsf{td})$**:** The generation algorithm takes as input the public parameter par and a string $\rho \in \mathcal{D}_{\mathsf{par}}$. It outputs a common reference string crs and a trapdoor td.

$\mathsf{Prove}(\mathsf{crs}, y, w) \to \pi$**:** The proving algorithm takes as input the crs, a statement y, and a witness w with $R_\rho(y, w) = 1$, and outputs a proof π.

$\mathsf{Ver}(\mathsf{crs}, y, \pi) \to 1/0$**:** The verification algorithm takes as input crs, a statement y, and a proof π, and outputs 1 or 0.

$\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, y) \to \pi$**:** The simulation algorithm takes as input crs, td, and a statement y (not necessarily in $\mathcal{L}_\rho$), and outputs a simulated proof π.

Perfect completeness. We say that a QA-NIZK Π satisfies perfect completeness, if for all $\lambda \in \mathbb{N}$, $\rho \in \mathcal{D}_{\mathsf{par}}$, (y, w) with $R_\rho(y, w) = 1$, and $(\mathsf{crs}, \mathsf{td}) \leftarrow\$ \mathsf{Gen}(\mathsf{par}, \rho)$, we have $\Pr[\mathsf{Ver}(\mathsf{crs}, y, \mathsf{Prove}(\mathsf{crs}, y, w)) = 1] = 1$.

Perfect zero-knowledge. We say that a QA-NIZK Π satisfies perfect zero-knowledge, if for all $\lambda \in \mathbb{N}$, $\rho \in \mathcal{D}_{\mathsf{par}}$, (y, w) with $R_\rho(y, w) = 1$, $(\mathsf{crs}, \mathsf{td}) \leftarrow\$ \mathsf{Gen}(\mathsf{par}, \rho)$, the two distributions $\{\mathsf{Prove}(\mathsf{crs}, y, w)\}$ and $\{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, y)\}$ are identical.

We define the simulation-soundness for a QA-NIZK argument.

Simulation-soundness. We say that a QA-NIZK Π satisfies the (unbounded) simulation-sound (USS) if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{USS}}_{\Pi,\mathcal{A}}(\lambda) := \Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1]$ is negligible, where Game $\mathsf{USS}^{\mathcal{A}}$ is defined in Figure 1.

We say that Π is one-time simulation-sound (OT-SS) if $\mathcal{A}$ can make at most one query to SIM (i.e., $Q_s = 1$). We denote the corresponding advantage function by $\mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{A}}(\lambda)$.

Remark 1 (Variants of definitions of simulation-soundness): Here, we use a stronger version of simulation-soundness used in [15], [33] that requires $(y^*, \pi^*) \notin Q_{\mathsf{sim}}$, rather than the weaker version used in [23], [34], [35] that only requires $y^* \notin Q_{\mathsf{sim}}$. As mentioned in [33], the weaker simulation-soundness is not sufficient to construct a CCA2-secure encryption scheme, because it does not prevent an adversary from sending a forged challenge ciphertext as a decryption query.

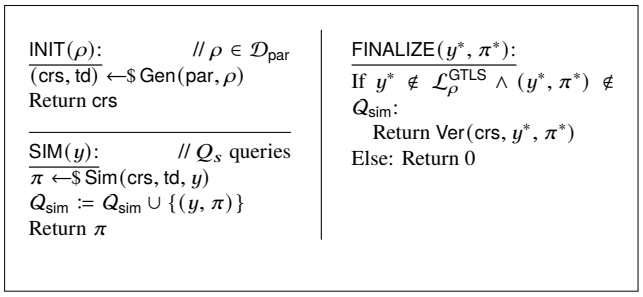


Fig. 1 USS security game for QA-NIZK

3.2 Construction: OT-SS QA-NIZK Argument for GTLS

Here, we show our OT-SS QA-NIZK argument for $\mathcal{L}^{\mathsf{GTLS}} := \{\mathcal{L}_\rho^{\mathsf{GTLS}}\}_{\rho\in\mathcal{D}_{\mathsf{par}}}$. Let $\mathcal{H} = \{\mathsf{H} : \{0,1\}^* \to \mathbb{Z}_q\}$ be a CR hash function family. Our QA-NIZK argument $\Pi_{\mathsf{OT\text{-}SS}} = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ is defined in Figure 2.[††]

[†] We can also easily construct *unbounded* simulation-sound one. Please refer to Appendix A for detail.

[††] As with the QA-NIZK argument proposed by Abe et al. [33],

$\underline{\mathsf{Gen}(\mathsf{par}, \rho = ([\mathbf{M}]_1, [\mathbf{M}'_1]_1, \ldots, [\mathbf{M}'_m]_1)):}$
$\mathsf{H} \leftarrow\!\$\, \mathcal{H}, \mathbf{A} \leftarrow\!\$\, \mathcal{D}_k \subset \mathbb{Z}_q^{(k+1)\times k}, \mathbf{K}^{(0)}, \mathbf{K}^{(1)} \leftarrow\!\$\, \mathbb{Z}_q^{n'\times(k+1)}$
$(\mathbf{Y}^{(0)}, \mathbf{Y}^{(1)}) := (\mathbf{K}^{(0)}\mathbf{A}, \mathbf{K}^{(1)}\mathbf{A})$
For $i = 1, \ldots, m$:
  $\mathbf{K}_i^{(0)}, \mathbf{K}_i^{(1)} \leftarrow\!\$\, \mathbb{Z}_q^{n\times(k+1)}$, $(\mathbf{Y}_i^{(0)}, \mathbf{Y}_i^{(1)}) := (\mathbf{K}_i^{(0)}\mathbf{A}, \mathbf{K}_i^{(1)}\mathbf{A})$
  $[\mathbf{P}_i^{(b)}]_1 := [\mathbf{M}^\top \mathbf{K}_i^{(b)} + \mathbf{M}_i'^\top \mathbf{K}^{(b)}]_1$ for $b \in \{0,1\}$
$\mathsf{crs} := (\{[\mathbf{P}_i^{(b)}]_1, [\mathbf{Y}_i^{(b)}]_2\}_{i,b}, \{[\mathbf{Y}^{(b)}]_2\}_b, [\mathbf{A}]_2, \mathsf{H})$
$\mathsf{td} := (\{\mathbf{K}_i^{(b)}\}_{i,b}, \{\mathbf{K}^{(b)}\}_b)$
Return $(\mathsf{crs}, \mathsf{td})$

$\underline{\mathsf{Prove}(\mathsf{crs}, ([\mathbf{c}]_1, \mathbf{x}), \mathbf{r}):}$ // $\mathbf{c} = \mathbf{M}_\mathbf{x}\mathbf{r}$
$\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x})$, $[\mathbf{P}_\mathbf{x}^{(b)}]_1 := [\sum_{i=1}^m x_i \mathbf{P}_i^{(b)}]_1$ for $b \in \{0,1\}$
$\pi := [(\mathbf{P}_\mathbf{x}^{(0)} + \tau\mathbf{P}_\mathbf{x}^{(1)})^\top \mathbf{r}]_1 \in \mathbb{G}_1^{k+1}$
Return π

$\underline{\mathsf{Ver}(\mathsf{crs}, ([\mathbf{c}]_1, \mathbf{x}), \pi = [\mathbf{u}]_1):}$
$\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x})$, $[\mathbf{Y}_\mathbf{x}^{(b)}]_2 := \left[\begin{matrix}\sum_{i=1}^m x_i \mathbf{Y}_i^{(b)} \\ \mathbf{Y}^{(b)}\end{matrix}\right]_2$ for $b \in \{0,1\}$
If $[\mathbf{u}]_1^\top \circ [\mathbf{A}]_2 = [\mathbf{c}]_1^\top \circ [\mathbf{Y}_\mathbf{x}^{(0)} + \tau\mathbf{Y}_\mathbf{x}^{(1)}]_2$: Return 1
Else: Return 0

$\underline{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, ([\mathbf{c}]_1, \mathbf{x})):}$
$\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x})$, $\mathbf{K}_\mathbf{x}^{(b)} := \left(\begin{matrix}\sum_{i=1}^m x_i \mathbf{K}_i^{(b)} \\ \mathbf{K}^{(b)}\end{matrix}\right)$ for $b \in \{0,1\}$
$\pi := [(\mathbf{K}_\mathbf{x}^{(0)} + \tau\mathbf{K}_\mathbf{x}^{(1)})^\top \mathbf{c}]_1 \in \mathbb{G}_1^{k+1}$
Return π

Fig. 2 Our OT-SS QA-NIZK argument $\Pi_{\mathsf{OT\text{-}SS}}$.

To prove OT-SS of $\Pi_{\mathsf{OT\text{-}SS}}$, we use the following lemma adapted from [23].

Lemma 2: Let $n, n', t, k \in \mathbb{N}$. Then, for any full rank matrix $\mathbf{M} \in \mathbb{Z}_q^{n\times t}$, any matrices $\mathbf{M}'_1, \ldots, \mathbf{M}'_m \in \mathbb{Z}_q^{n'\times t}$ and $\mathbf{A} \in \mathbb{Z}_q^{(k+1)\times k}$, and any (possibly unbounded) adversary $\mathcal{A}$, we have $\Pr[\mathsf{Core}^{\mathcal{A}}_{\mathsf{OT\text{-}SS}} \Rightarrow 1] \le 1/q$, where Game $\mathsf{Core}_{\mathsf{OT\text{-}SS}}$ is defined in Figure 3.

Proof. To prove the lemma, fix matrices $\mathbf{M} \in \mathbb{Z}_q^{n\times t}$, $\mathbf{M}'_1, \ldots, \mathbf{M}'_m \in \mathbb{Z}_q^{n'\times t}$, $\mathbf{A} \in \mathbb{Z}_q^{(k+1)\times k}$, and fix a non-zero vector $\hat{\mathbf{a}} \notin \mathsf{Span}(\mathbf{A})$. For any $\mathbf{x}^* \in \mathbb{Z}_q^m$ and $\mathbf{c}^* \in \mathbb{Z}_q^{n+n'}$ such that $\mathbf{c}^* \notin \mathsf{Span}(\mathbf{M}_{\mathbf{x}^*})$, there exist vectors $\mathbf{m}_1^\perp, \ldots, \mathbf{m}_m^\perp \in \mathbb{Z}_q^n$ and $\mathbf{m}^\perp \in \mathbb{Z}_q^{n'}$ such that

$$\begin{pmatrix} \mathbf{M}^\top & \mathbf{O} & \cdots & \mathbf{O} & \mathbf{M}_1'^\top \\ \mathbf{O} & \mathbf{M}^\top & \cdots & \mathbf{O} & \mathbf{M}_2'^\top \\ \vdots & & \ddots & & \vdots \\ \mathbf{O} & \cdots & \mathbf{O} & \mathbf{M}^\top & \mathbf{M}_m'^\top \\ x_1^*\overline{\mathbf{c}^*}^\top & x_2^*\overline{\mathbf{c}^*}^\top & \cdots & x_m^*\overline{\mathbf{c}^*}^\top & \underline{\mathbf{c}^*}^\top \end{pmatrix} \begin{pmatrix} \mathbf{m}_1^\perp \\ \mathbf{m}_2^\perp \\ \vdots \\ \mathbf{m}_m^\perp \\ \mathbf{m}^\perp \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{4}$$

our construction can be easily extended to a *tag-based* QA-NIZK argument by adding the label lbl to the input of hash function. Thus, our construction can be used in all the applications that require tag-based QA-NIZK arguments.

$\underline{\mathsf{INIT}_{\mathsf{Core}}:}$
$\mathbf{K}^{(0)}, \mathbf{K}^{(1)} \leftarrow\!\$\, \mathbb{Z}_q^{n'\times(k+1)}$, $(\mathbf{Y}^{(0)}, \mathbf{Y}^{(1)}) := (\mathbf{K}^{(0)}\mathbf{A}, \mathbf{K}^{(1)}\mathbf{A})$
For $i = 1, \ldots, m$:
  $\mathbf{K}_i^{(0)}, \mathbf{K}_i^{(1)} \leftarrow\!\$\, \mathbb{Z}_q^{n\times(k+1)}$, $(\mathbf{Y}_i^{(0)}, \mathbf{Y}_i^{(1)}) := (\mathbf{K}_i^{(0)}\mathbf{A}, \mathbf{K}_i^{(1)}\mathbf{A})$
  $\mathbf{P}_i^{(b)} := \mathbf{M}^\top \mathbf{K}_i^{(b)} + \mathbf{M}_i'^\top \mathbf{K}^{(b)}$ for $b \in \{0,1\}$
$\mathsf{crs}_{\mathsf{Core}} := (\{\mathbf{P}_i^{(b)}, \mathbf{Y}_i^{(b)}\}_{i,b}, \{\mathbf{Y}^{(b)}\}_b, \mathbf{A})$
Return $\mathsf{crs}_{\mathsf{Core}}$

$\underline{\mathsf{EVAL}_{\mathsf{Core}}(\mathbf{x}, \tau):}$ // one query
$\mathbf{K}_\mathbf{x}^{(b)} := \left(\begin{matrix}\sum_{i=1}^m x_i \mathbf{K}_i^{(b)} \\ \mathbf{K}^{(b)}\end{matrix}\right)$ for $b \in \{0,1\}$
Return $\mathbf{K}_\mathbf{x}^{(0)} + \tau\mathbf{K}_\mathbf{x}^{(1)}$

$\underline{\mathsf{FINALIZE}_{\mathsf{Core}}([\mathbf{u}^*]_1, ([\mathbf{c}^*]_1, \mathbf{x}^*), \tau^*):}$
If $[\mathbf{c}^*]_1 \notin \mathsf{Span}([\mathbf{M}_{\mathbf{x}^*}]_1) \wedge \tau^* \neq \tau \wedge [\mathbf{u}^*]_1 = (\mathbf{K}_{\mathbf{x}^*}^{(0)} + \tau\mathbf{K}_{\mathbf{x}^*}^{(1)})^\top [\mathbf{c}^*]_1$:
  Return 1
Else: Return 0

Fig. 3 Game $\mathsf{Core}_{\mathsf{OT\text{-}SS}}$ for defining Lemma 2

since $\mathbf{c}^* \notin \mathsf{Span}(\mathbf{M}_{\mathbf{x}^*})$. By setting $\mathbf{K}^{(b)} := \hat{\mathbf{K}}^{(b)} + s\mathbf{m}^\perp\mathbf{a}^\perp$ and $\mathbf{K}_i^{(b)} := \hat{\mathbf{K}}_i^{(b)} + s\mathbf{m}_i^\perp\mathbf{a}^\perp$ for $b \in \{0,1\}$, where $s \leftarrow\!\$\, \mathbb{Z}_q$, $\hat{\mathbf{K}}^{(0)}, \hat{\mathbf{K}}^{(1)} \leftarrow\!\$\, \mathbb{Z}_q^{n'\times(k+1)}$, $\hat{\mathbf{K}}_i^{(0)}, \hat{\mathbf{K}}_i^{(1)} \leftarrow\!\$\, \mathbb{Z}_q^{n\times(k+1)}$ and $\mathbf{a}^\perp \in \ker(\mathbf{A})$, we can see that the following two distributions

$$(\mathsf{crs}_{\mathsf{Core}}, \mathbf{K}_\mathbf{x}^{(0)} + \tau\mathbf{K}_\mathbf{x}^{(1)}, \hat{\mathbf{a}}^\top(\mathbf{K}_{\mathbf{x}^*}^{(0)} + \tau\mathbf{K}_{\mathbf{x}^*}^{(1)})^\top \mathbf{c}^*) \text{ and}$$
$$(\mathsf{crs}_{\mathsf{Core}}, \mathbf{K}_\mathbf{x}^{(0)} + \tau\mathbf{K}_\mathbf{x}^{(1)}, u)$$

are the same, where $u \leftarrow\!\$\, \mathbb{Z}_q$.

By a standard argument (e.g., complexity leveraging), this means that the two distributions are the same even if $\mathbf{x}^*$ and $\mathbf{c}^*$ are adaptively chosen after seeing $(\mathsf{crs}_{\mathsf{Core}}, \mathbf{K}_\mathbf{x}^{(0)} + \tau\mathbf{K}_\mathbf{x}^{(1)})$. Hence, for any adversary $\mathcal{A}$, we have $\Pr[\mathsf{Core}^{\mathcal{A}}_{\mathsf{OT\text{-}SS}} \Rightarrow 1] \le 1/q$ since $\hat{\mathbf{a}}^\top(\mathbf{K}_{\mathbf{x}^*}^{(0)} + \tau\mathbf{K}_{\mathbf{x}^*}^{(1)})^\top \mathbf{c}^*$ is uniformly random from the $\mathcal{A}$'s view point. □

Theorem 1: $\Pi_{\mathsf{OT\text{-}SS}}$ defined in Figure 2 has perfect completeness and perfect zero-knowledge. Furthermore, if the $\mathcal{D}_k$-KerMDH problem in $\mathbb{G}_2$ is hard and $\mathcal{H}$ is a CR hash function family, then $\Pi_{\mathsf{OT\text{-}SS}}$ has one-time simulation-soundness.

Proof. Perfect completeness and perfect zero-knowledge follow readily from the fact that

$$\begin{aligned}
&\left(\mathbf{P}_\mathbf{x}^{(0)} + \tau\mathbf{P}_\mathbf{x}^{(1)}\right)^\top \\
&= \sum_{i=1}^m \left(x_i\mathbf{P}_i^{(0)} + \tau x_i\mathbf{P}_i^{(1)}\right)^\top \\
&= \sum_{i=1}^m \left(x_i(\mathbf{K}_i^{(0)\top}\mathbf{M} + \mathbf{K}^{(0)\top}\mathbf{M}'_i) + \tau x_i(\mathbf{K}_i^{(1)\top}\mathbf{M} + \mathbf{K}^{(1)\top}\mathbf{M}'_i)\right) \\
&= \begin{pmatrix}\sum_{1=1}^m x_i\mathbf{K}_i^{(0)} \\ \mathbf{K}^{(0)}\end{pmatrix}^\top \begin{pmatrix}\mathbf{M} \\ \sum_{1=1}^m x_i\mathbf{M}'_i\end{pmatrix} + \tau\begin{pmatrix}\sum_{1=1}^m x_i\mathbf{K}_i^{(1)} \\ \mathbf{K}^{(1)}\end{pmatrix}^\top \begin{pmatrix}\mathbf{M} \\ \sum_{1=1}^m x_i\mathbf{M}'_i\end{pmatrix} \\
&= \mathbf{K}_\mathbf{x}^{(0)\top}\mathbf{M}_\mathbf{x} + \tau\mathbf{K}_\mathbf{x}^{(1)\top}\mathbf{M}_\mathbf{x}
\end{aligned}$$

$\underline{\mathsf{INIT}(\rho)}$: // G_0–G_2
$\mathsf{H} \leftarrow\$ \mathcal{H}$, $\mathbf{A} \leftarrow\$ \mathcal{D}_k \subset \mathbb{Z}_q^{(k+1)\times k}$, $\mathbf{K}^{(0)}, \mathbf{K}^{(1)} \leftarrow\$ \mathbb{Z}_q^{n'\times(k+1)}$
$(\mathbf{Y}^{(0)}, \mathbf{Y}^{(1)}) := (\mathbf{K}^{(0)}\mathbf{A}, \mathbf{K}^{(1)}\mathbf{A})$
For $i = 1, \ldots, m$:
  $\mathbf{K}_i^{(0)}, \mathbf{K}_i^{(1)} \leftarrow\$ \mathbb{Z}_q^{n\times(k+1)}$, $(\mathbf{Y}_i^{(0)}, \mathbf{Y}_i^{(1)}) := (\mathbf{K}_i^{(0)}\mathbf{A}, \mathbf{K}_i^{(1)}\mathbf{A})$
  $[\mathbf{P}_i^{(b)}]_1 := [\mathbf{M}^\top \mathbf{K}_i^{(b)} + \mathbf{M}_i'^\top \mathbf{K}^{(b)}]_1$ for $b \in \{0,1\}$
$\mathsf{crs} := (\{[\mathbf{P}_i^{(b)}]_1, [\mathbf{Y}_i^{(b)}]_2\}_{i,b}, \{[\mathbf{Y}^{(b)}]_2\}_b, [\mathbf{A}]_2, \mathsf{H})$
Return crs

$\underline{\mathsf{SIM}([\mathbf{c}]_1, \mathbf{x})}$: // G_0–G_2
$\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x})$, $\mathbf{K}_{\mathbf{x}}^{(b)} := \begin{pmatrix} \sum_{i=1}^m x_i \mathbf{K}_i^{(b)} \\ \mathbf{K}^{(b)} \end{pmatrix}$ for $b \in \{0,1\}$
$\pi := (\mathbf{K}_{\mathbf{x}}^{(0)} + \tau \mathbf{K}_{\mathbf{x}}^{(1)})^\top [\mathbf{c}]_1 \in \mathbb{G}_1^{k+1}$
Return π

$\underline{\mathsf{FINALIZE}(([\mathbf{c}^*]_1, \mathbf{x}^*), \pi^* = [\mathbf{u}^*]_1)}$: // G_0, G_1, G_2
If $([\mathbf{c}^*]_1, \mathbf{x}^*) \in \mathcal{L}_\rho^{\mathsf{GTLS}}$ or $(([\mathbf{c}^*]_1, \mathbf{x}^*), \pi^*) = (([\mathbf{c}]_1, \mathbf{x}), \pi)$:
  Return 0
$\tau^* := \mathsf{H}([\mathbf{c}^*]_1, \mathbf{x}^*)$, $[\mathbf{Y}_{\mathbf{x}}^{(b)}]_2 := \left[\begin{matrix} \sum_{i=1}^m x_i \mathbf{Y}_i^{(b)} \\ \mathbf{Y}^{(b)} \end{matrix} \right]_2$ for $b \in \{0,1\}$
If $\tau^* = \tau$: Return 0
If $[\mathbf{u}^*]_1^\top \circ [\mathbf{A}]_2 = [\mathbf{c}^*]_1^\top \circ [\mathbf{Y}_{\mathbf{x}^*}^{(0)} + \tau \mathbf{Y}_{\mathbf{x}^*}^{(1)}]_2$: Return 1
If $[\mathbf{u}^*]_1 = (\mathbf{K}_{\mathbf{x}^*}^{(0)} + \tau^* \mathbf{K}_{\mathbf{x}^*}^{(1)})^\top [\mathbf{c}^*]_1$: Return 1
Else: Return 0

Fig. 4 Games G_0, G_1, and G_2 for the proof of Theorem 1. In each procedure, a solid (dotted, gray) frame indicates that the command is only executed in the game marked by a solid (dotted, gray) frame.

$$= (\mathbf{K}_{\mathbf{x}}^{(0)\top} + \tau \mathbf{K}_{\mathbf{x}}^{(1)\top}) \mathbf{M}_{\mathbf{x}},$$

and for all $\mathbf{c} = \mathbf{M}_{\mathbf{x}}\mathbf{r}$, we have

$$\left(\mathbf{P}_{\mathbf{x}}^{(0)} + \tau \mathbf{P}_{\mathbf{x}}^{(1)}\right)^\top \mathbf{r} = \left(\mathbf{K}_{\mathbf{x}}^{(0)\top} + \tau \mathbf{K}_{\mathbf{x}}^{(1)\top}\right) \mathbf{M}_{\mathbf{x}} \mathbf{r} = \left(\mathbf{K}_{\mathbf{x}}^{(0)} + \tau \mathbf{K}_{\mathbf{x}}^{(1)}\right)^\top \mathbf{c},$$

where $\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x})$.

Next, we will prove that $\Pi_{\mathsf{OT\text{-}SS}}$ has OT-SS. We will show that for any adversary $\mathcal{A}$, there exists adversaries $\mathcal{B}$ and $\hat{\mathcal{B}}$ with

$$\mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi_{\mathsf{OT\text{-}SS}},\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{KerMDH}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) + \mathsf{Adv}^{\mathsf{CR}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda) + 1/q. \tag{5}$$

We bound the advantage of $\mathcal{A}$ via a sequence of games defined in Figure 4. G_0 is the real OT-SS game for QA-NIZK as defined in Figure 1.

Lemma 3 (G_0): $\Pr[\mathsf{OT\text{-}SS}^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1]$.

Lemma 4 (G_0 to G_1): There is an adversary $\mathcal{B}$ that solves the $\mathcal{D}_k$-KerMDH problem in $\mathbb{G}_2$ with $\mathsf{Adv}^{\mathsf{KerMDH}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) \ge |\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]|$.

Proof. G_1 is identical to G_0 unless $\mathcal{A}$ queries FINALIZE with $(([\mathbf{c}^*], \mathbf{x}^*), [\mathbf{u}^*]_1)$ such that $[\mathbf{u}^*]_1 - (\mathbf{K}_{\mathbf{x}^*}^{(0)} + \tau^* \mathbf{K}_{\mathbf{x}^*}^{(1)})^\top [\mathbf{c}^*]_1$ is a non-zero vector in $\ker(\mathbf{A})$, which is a solution of the $\mathcal{D}_k$-KerMDH problem for $\mathbf{A}$. Hence, we have $|\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]| \le \mathsf{Adv}^{\mathsf{KerMDH}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda)$. □

$\underline{\mathsf{INIT}(\rho)}$:
$\mathsf{crs}_{\mathsf{Core}} \leftarrow\$ \mathsf{INIT}_{\mathsf{Core}}$, $\mathsf{H} \leftarrow\$ \mathcal{H}$
Parse $\mathsf{crs}_{\mathsf{Core}} =: (\{\mathbf{P}_i^{(b)}, \mathbf{Y}_i^{(b)}\}_{i,b}, \{\mathbf{Y}^{(b)}\}_b, \mathbf{A})$
$\mathsf{crs} := (\{[\mathbf{P}_i^{(b)}]_1, [\mathbf{Y}_i^{(b)}]_2\}_{i,b}, \{[\mathbf{Y}^{(b)}]_2\}_b, [\mathbf{A}]_2, \mathsf{H})$
Return crs

$\underline{\mathsf{SIM}([\mathbf{c}]_1, \mathbf{x})}$:
$\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x})$, $\mathbf{K}_{\mathbf{x}}^{(0)} + \tau \mathbf{K}_{\mathbf{x}}^{(1)} \leftarrow\$ \mathsf{EVAL}_{\mathsf{Core}}(\mathbf{x}, \tau)$
$\pi := (\mathbf{K}_{\mathbf{x}}^{(0)} + \tau \mathbf{K}_{\mathbf{x}}^{(1)})^\top [\mathbf{c}]_1 \in \mathbb{G}_1^{k+1}$
Return π

$\underline{\mathsf{FINALIZE}(([\mathbf{c}^*]_1, \mathbf{x}^*), \pi^* = [\mathbf{u}^*]_1)}$:
If $([\mathbf{c}^*]_1, \mathbf{x}^*) \in \mathcal{L}_\rho^{\mathsf{GTLS}}$ or $(([\mathbf{c}^*]_1, \mathbf{x}^*), \pi^*) = (([\mathbf{c}]_1, \mathbf{x}), \pi)$:
  Return 0
$\tau^* := \mathsf{H}([\mathbf{c}^*]_1, \mathbf{x}^*)$
If $\tau^* = \tau$: Return 0
Return $\mathsf{FINALIZE}_{\mathsf{Core}}([\mathbf{u}^*]_1, ([\mathbf{c}^*]_1, \mathbf{x}^*), \tau^*)$

Fig. 5 Algorithm $\mathcal{B}'$ for the proof of Lemma 6 with oracles $\mathsf{INIT}_{\mathsf{Core}}$, $\mathsf{EVAL}_{\mathsf{Core}}$, and $\mathsf{FINALIZE}_{\mathsf{Core}}$ defined in Figure 3. The oracle calls are highlighted with gray.

Lemma 5 (G_1 to G_2): There is an adversary $\hat{\mathcal{B}}$ breaking the collision resistance of $\mathcal{H}$ with $\mathsf{Adv}^{\mathsf{CR}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda) \ge |\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]|$.

Proof. The difference between G_1 and G_2 happens when $\mathcal{A}$ queries FINALIZE with $(([\mathbf{c}^*]_1, \mathbf{x}^*), \pi^*)$ such that $(([\mathbf{c}^*]_1, \mathbf{x}^*), \pi^*) \ne (([\mathbf{c}]_1, \mathbf{x}), \pi)$ and $\tau^* = \tau$. To bound this, we consider the following cases:

- $([\mathbf{c}^*]_1, \mathbf{x}^*) = ([\mathbf{c}]_1, \mathbf{x})$ and $\pi^* \ne \pi$. For $([\mathbf{c}^*]_1, \mathbf{x}^*) = ([\mathbf{c}]_1, \mathbf{x})$, only $(\mathbf{K}_{\mathbf{x}^*}^{(0)} + \tau^* \mathbf{K}_{\mathbf{x}^*}^{(1)})^\top [\mathbf{c}^*]_1 = (\mathbf{K}_{\mathbf{x}}^{(0)} + \tau \mathbf{K}_{\mathbf{x}}^{(1)})^\top [\mathbf{c}]_1 = \pi$ is accepted in both G_1 and G_2. Hence, in this case, the FINALIZE will output 0 in both games since $\pi^* \ne \pi$.
- $([\mathbf{c}^*]_1, \mathbf{x}^*) \ne ([\mathbf{c}]_1, \mathbf{x})$ and $\tau^* = \tau$. In this case, we can break the collision-resistance of $\mathcal{H}$. Hence, we can bound the probability of this case by $\mathsf{Adv}^{\mathsf{CR}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda)$.

Therefore, we have $|\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]| \le \mathsf{Adv}^{\mathsf{CR}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda)$. □

Lemma 6 (G_2): $\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] \le 1/q$.

Proof. To bound this probability, we consider the algorithm $\mathcal{B}'$ defined in Figure 5. Clearly, if the oracle access of $\mathcal{B}'$ is from $\mathsf{Core}_{\mathsf{OT\text{-}SS}}$, then $\mathcal{B}$ perfectly simulates G_2. Thus, we have $\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{Core}^{\mathcal{B}'}_{\mathsf{OT\text{-}SS}} \Rightarrow 1]$. From Lemma 2, we have $\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] \le 1/q$. □

From Lemmas 3 to 6, we obtain Eq. (5) □

3.3 GTLS Expressed by Linear Maps

We describe that a QA-NIZK argument for GTLS implies one for GTLS expressed by linear maps. Let $\rho := ([\mathbf{M}]_1, [\mathbf{M}'_1]_1, \ldots, [\mathbf{M}'_m]_1) \in \mathbb{G}_1^{n\times t} \times (\mathbb{G}_1^{n'\times t})^m$. For any $\mathbb{Z}_q$-linear map $L : (\mathbb{Z}_q^{n'\times t})^m \to \mathbb{Z}_q^{n'm'\times t}$, we define

$$\hat{\mathcal{L}}_\rho^{\mathsf{GTLS}} := \left\{([\mathbf{c}]_1, L) \mid \exists \mathbf{r} \in \mathbb{Z}_q^t \text{ s.t. } \mathbf{c} = \mathbf{M}_L \mathbf{r}\right\},$$

where $\mathbf{M}_L := \left(L(\mathbf{M}'_1, \ldots, \mathbf{M}'_m)\right)$ Note that a linear subspace to which $\mathbf{c}$ should belong depends on a linear map L, but not on a vector $\mathbf{x}$. We can see that our QA-NIZK arguments support the above language $\hat{\mathcal{L}}_\rho^{\mathsf{GTLS}}$ as follows: If we express L as $\mathbf{L} = (l_{i,j}) \in \mathbb{Z}_q^{m\times m'}$, we have

$$\begin{aligned} L(\mathbf{M}'_1, \ldots, \mathbf{M}'_m) &= \begin{pmatrix} \sum_{i=1}^m l_{i,1}\mathbf{M}'_i \\ \vdots \\ \sum_{i=1}^m l_{i,m'}\mathbf{M}'_i \end{pmatrix} \\ &= \sum_{i=1}^m l_{i,1}\begin{pmatrix} \mathbf{M}'_i \\ \vdots \\ \mathbf{O} \end{pmatrix} + \cdots + \sum_{i=1}^m l_{i,m'}\begin{pmatrix} \mathbf{O} \\ \vdots \\ \mathbf{M}'_i \end{pmatrix}, \end{aligned}$$

where $\mathbf{O} \in \mathbb{Z}_q^{n'\times t}$ is the matrix whose coordinates are all zero. Therefore, we can appropriately determine $\hat{\mathbf{M}}'_{1,1}, \ldots, \hat{\mathbf{M}}'_{m,m'} \in \mathbb{Z}_q^{m'n'\times t}$ from $\mathbf{M}'_1, \ldots, \mathbf{M}'_m$ that satisfy $L(\mathbf{M}'_1, \ldots, \mathbf{M}'_m) = \sum_{i=1}^m \sum_{j=1}^{m'} l_{i,j}\hat{\mathbf{M}}'_{i,j}$, and we have

$$\begin{aligned} \hat{\mathcal{L}}_\rho^{\mathsf{GTLS}} &= \left\{ ([\mathbf{c}]_1, \mathbf{l}) \;\middle|\; \exists \mathbf{r} \in \mathbb{Z}_q^t \text{ s.t. } \mathbf{c} = \left(\begin{matrix} \mathbf{M} \\ \sum_{i=1}^m \sum_{j=1}^{m'} l_{i,j}\hat{\mathbf{M}}'_{i,j} \end{matrix}\right)\mathbf{r} \right\} \\ &= \mathcal{L}_{\hat{\rho}}^{\mathsf{GTLS}} \in \mathcal{L}^{\mathsf{GTLS}}, \end{aligned}$$

where $\hat{\rho} := ([\mathbf{M}]_1, [\hat{\mathbf{M}}'_{1,1}]_1, \ldots, [\hat{\mathbf{M}}'_{m,m'}]_1) \in \mathbb{G}_1^{n\times t} \times (\mathbb{G}_1^{n'\times t})^{mm'}$ and $\mathbf{l} := (l_{1,1}, \ldots, l_{m,m'})^\top \in \mathbb{Z}_q^{mm'}$.

4. Leakage-Resilient CCA2-Secure Attribute-Based Encryption

In this section, we present the first leakage-resilient CCA2 (LR-CCA2) secure attribute-based key encapsulation (ABKEM)[†] scheme. Our LR-CCA2-secure ABKEM scheme is obtained by combining the LR-CPA-secure ABE scheme by Zhang et al. [11] and our QA-NIZK argument in Section 3. Similar to the scheme of [11], our LR-CCA2-secure scheme is resilient to the leakage of both master and user's secret key, and works in the CML model.

In Section 4.1, we first give the definition of ABKEM and its security model. In Section 4.2, we then recall the notion of leakage-resilient predicate encodings that is the main building block of the ABE scheme of [11]. In Section 4.3, we finally provide the construction of our ABKEM scheme.

[†]Here, we only focus on ABKEMs, since, even in the leakage-resilient setting, a CCA2-secure ABKEM scheme can be transformed to a CCA2-secure ABE scheme efficiently and securely by using a symmetric encryption scheme. We can prove this by adapting the techniques from [36] in a straightforward manner.

4.1 Definition

Here, we provide the definition of ABKEM and its security model.

Syntax. An ABKEM scheme for a predicate $P : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ consists of the following PPT algorithms.

$\mathsf{Setup}(1^\lambda, \mathcal{X}, \mathcal{Y}) \to (\mathsf{mpk}, \mathsf{msk})$**:** The setup algorithm takes as input a security parameter 1^λ, the ciphertext attribute universe $\mathcal{X}$ and the key attribute universe $\mathcal{Y}$, and outputs a master public key mpk and a master secret key msk. We assume that mpk implicitly defines an encapsulated key space $\mathcal{K}$.

$\mathsf{KGen}(\mathsf{msk}, y) \to \mathsf{sk}_y$**:** The key generation algorithm takes as input the master secret key msk and a key attribute $y \in \mathcal{Y}$, and outputs a secret key sk_y.

$\mathsf{Encap}(\mathsf{mpk}, x) \to (\mathsf{K}, \mathsf{ct}_x)$**:** The encapsulation algorithm takes as input the master public key mpk and an attribute $x \in \mathcal{X}$, and outputs an encapsulated key $\mathsf{K} \in \mathcal{K}$ together with a ciphertext ct_x.

$\mathsf{Decap}((\mathsf{sk}_y, y), \mathsf{ct}_x) \to \mathsf{K}$ **or** $\perp$**:** The decapsulation algorithm takes as input the secret key sk_y with the corresponding attribute y and the ciphertext ct_x, and outputs either an encapsulated key K or $\perp$ (indicating the ciphertext is invalid).

In the CML model, we use the following key update algorithm in addition to the above algorithms.

$\mathsf{UpdateMSK}(\mathsf{mpk}, \mathsf{msk}) \to \mathsf{msk}'$**:** The master secret key update algorithms takes as input the master public key mpk and the master secret key msk, and outputs a re-randomized master secret key msk'.

$\mathsf{UpdateSK}(\mathsf{mpk}, \mathsf{sk}_y, y) \to \mathsf{sk}'_y$**:** The secret key update algorithms takes as input the master public key mpk and a secret key sk_y with the corresponding attribute y, and outputs a re-randomized secret key sk'_y.

Correctness. For all $\lambda \in \mathbb{N}$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow\!\$\, \mathsf{Setup}(1^\lambda, \mathcal{X}, \mathcal{Y})$, attributes $(x, y) \in \mathcal{X} \times \mathcal{Y}$ such that $P(x, y) = 1$, $\mathsf{sk}_y \leftarrow\!\$\, \mathsf{KGen}(\mathsf{msk}, y)$, and $(\mathsf{K}, \mathsf{ct}_x) \leftarrow\!\$\, \mathsf{Encap}(\mathsf{mpk}, x)$, we have $\Pr[\mathsf{Decap}((\mathsf{sk}_y, y), \mathsf{ct}_x) = \mathsf{K}] = 1$. Furthermore, in the CML model, the above property is also satisfied even if we use a re-randomized master secret key or a secret key.

We define the LR-CCA2-security of ABKEM in the CML model. In the CML model [5], [7], there is a notion of time periods and secret keys are updated at the end of each time period. An adversary is allowed to obtain a bounded leakage of secret keys in each time period, but there is no bound on the overall leakage.

LR-CCA2-Security for ABKEM in the CML model. Let $\ell_{\mathsf{msk}} = \ell_{\mathsf{msk}}(\lambda)$ and $\ell_{\mathsf{sk}} = \ell_{\mathsf{sk}}(\lambda)$ be leakage bounds for a master key and secret keys, respectively. An ABKEM scheme

INIT:
$b \leftarrow\$ \{0,1\}$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow\$ \mathsf{Setup}(1^\lambda, \mathcal{X}, \mathcal{Y})$
$Q_T := \{(0, \epsilon, \mathsf{msk}, 0)\}$
Return mpk

CREATE(h, y):
Find $(h, y', \mathsf{sk}_y, L) \in Q_T$
If $y' = \epsilon$ (i.e., $\mathsf{sk}_\epsilon = \mathsf{msk}'$):
  $\mathsf{sk}_y \leftarrow\$ \mathsf{KGen}(\mathsf{msk}', y)$
  $Q_T := Q_T \cup \{(H+1, y, \mathsf{sk}_y, 0)\}$, $H := H+1$
Return $\perp$

REVEAL(h):
Find $(h, y, \mathsf{sk}_y, L) \in Q_T$
If $y \in Q_R$ s.t. $P(x^*, y) = 1$: Return $\perp$
If $y \neq \epsilon$:
  $Q_R := Q_R \cup \{y\}$
  Return sk_y
Else: Return $\perp$

LEAK(h, f):
If flg = 1: Return $\perp$
Find $(h, y, \mathsf{sk}_y, L) \in Q_T$
If $y \neq \epsilon$ and $L + |f(\mathsf{sk}_y)| \leq \ell_{\mathsf{sk}}$:
  $L := L + |f(\mathsf{sk}_y)|$
  Return $f(\mathsf{sk}_y)$
If $y = \epsilon$ (i.e., $\mathsf{sk}_\epsilon = \mathsf{msk}$) and $L + |f(\mathsf{msk})| \leq \ell_{\mathsf{msk}}$:
  $L := L + |f(\mathsf{msk})|$
  Return $f(\mathsf{msk})$
Return $\perp$

UPDATE(h):
Find $(h, y, \mathsf{sk}_y, L) \in Q_T$
If $y = \epsilon$ (i.e., $\mathsf{sk}_\epsilon = \mathsf{msk}$):
  $\mathsf{msk}' \leftarrow\$ \mathsf{UpdateMSK}(\mathsf{mpk}, \mathsf{msk})$
  $Q_T := Q_T \cup \{(H+1, \epsilon, \mathsf{msk}', 0)\}$, $H := H+1$
Else:
  $\mathsf{sk}'_y \leftarrow\$ \mathsf{UpdateSK}(\mathsf{mpk}, \mathsf{sk}_y, y)$
  $Q_T := Q_T \cup \{(H+1, y, \mathsf{sk}'_y, 0)\}$, $H := H+1$
Return $\perp$

DECAP(h, ct_x):
Find $(h, y, \mathsf{sk}_y, L) \in Q_T$
If $y = \epsilon$ (i.e., $\mathsf{sk}_\epsilon = \mathsf{msk}$): Return $\perp$
If $\mathsf{ct}_x \neq \mathsf{ct}^*$ or $R(x^*, y) = 0$: Return $\mathsf{Decap}((\mathsf{sk}_y, y), \mathsf{ct}_x)$
Return $\perp$

CHAL(x^*): // one query
flg := 1
If $y \in Q_R$ s.t. $P(x^*, y) = 1$: Return $\perp$
$(\mathsf{ct}^*, \mathsf{K}^*_0) \leftarrow\$ \mathsf{Encap}(\mathsf{mpk}, x^*)$, $\mathsf{K}^*_1 \leftarrow\$ \mathcal{K}$
Return $(\mathsf{ct}^*, \mathsf{K}^*_b)$

FINALIZE(b'):
Return $(b' = b)$

Fig. 6 Security game $\mathsf{LR\text{-}CCA}_{\mathsf{CML}}$.

ABKEM is $(\ell_{\mathsf{msk}}, \ell_{\mathsf{sk}})$-LR-CCA2-secure if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{LR\text{-}CCA}}_{\mathsf{ABKEM},\mathcal{A}}(\lambda) := \left|\Pr[\mathsf{LR\text{-}CCA}^{\mathcal{A}}_{\mathsf{CML}} \Rightarrow 1] - 1/2\right|$ is negligible, where Game $\mathsf{LR\text{-}CCA}_{\mathsf{CML}}$ is defined as in Figure 6.

Remark 2 (On the handle in Figure 6): The first component in each entry of Q_T is called "handle," which is introduced to specify a master secret key/user's secret key that is updated every time period. The handle of the original master secret key is 0.

4.2 Leakage-Resilient Predicate Encodings

Here, we recall the notion of leakage-resilient predicate encodings (LRPE) introduced by Zhang et al. [11] to construct a leakage-resilient ABE scheme.

Definitions. Let $P : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ be a predicate. An LRPE for P is a tuple of deterministic algorithms $(\mathsf{sE}, \mathsf{mE}, \mathsf{mkE}, \mathsf{rE}, \mathsf{rkE}, \mathsf{sD}, \mathsf{rD})$ as follows: $\mathsf{sE} : \mathcal{X} \times \mathbb{Z}_q^n \to \mathbb{Z}_q^{n_s}$, $\mathsf{mE} : \mathbb{Z}_q^{n_z} \times \mathbb{Z}_q^n \to \mathbb{Z}_q^{n_m}$, $\mathsf{mkE} : \mathbb{Z}_q^{n_z} \times \mathbb{Z}_q \to \mathbb{Z}_q^{n_m}$, $\mathsf{rE} : \mathcal{Y} \times \mathbb{Z}_q^{n_z} \times \mathbb{Z}_q^n \to \mathbb{Z}_q^{n_r}$, $\mathsf{rkE} : \mathcal{Y} \times \mathbb{Z}_q^{n_z} \times \mathbb{Z}_q \to \mathbb{Z}_q^{n_r}$, $\mathsf{sD} : \mathcal{X} \times \mathcal{Y} \times \mathbb{Z}_q^{n_z} \times \mathbb{Z}_q^{n_s} \to \mathbb{Z}_q$, $\mathsf{rD} : \mathcal{X} \times \mathcal{Y} \times \mathbb{Z}_q^{n_z} \times \mathbb{Z}_q^{n_r} \to \mathbb{Z}_q$ for some $n, n_s, n_r, n_m, n_z \in \mathbb{N}$. We require that an LRPE satisfies linearity, α-reconstruction, α-privacy, α-leakage-resilient, delegable, and re-randomizable. We highlight only linearity, delegable, and re-randomizable that will be used in the following. Please refer to [11] for more details.

(linearity): For all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and $\mathbf{z} \in \mathbb{Z}_q^{n_z}$, the functions $\mathsf{sE}(x, \cdot)$, $\mathsf{mE}(\mathbf{z}, \cdot)$, $\mathsf{mkE}(\mathbf{z}, \cdot)$, $\mathsf{rE}(y, \mathbf{z}, \cdot)$, $\mathsf{rkE}(y, \mathbf{z}, \cdot)$, $\mathsf{sD}(x, y, \mathbf{z}, \cdot)$ and $\mathsf{rD}(x, y, \mathbf{z}, \cdot)$ are $\mathbb{Z}_q$-linear.

(delegable): For all $\alpha \in \mathbb{Z}_q$, $\mathbf{z} \in \mathbb{Z}_q^{n_z}$, $\mathbf{w} \in \mathbb{Z}_q^n$ and $y \in \mathcal{Y}$, there exists a linear map $\mathsf{D}(y, \cdot) : \mathsf{mkE}(\mathbf{z}, \alpha) + \mathsf{mE}(\mathbf{z}, \mathbf{w}) \mapsto \mathsf{rkE}(y, \mathbf{z}, \alpha) + \mathsf{rE}(y, \mathbf{z}, \mathbf{w})$.

(re-randomizable): For all $\alpha \in \mathbb{Z}_q$, $\mathbf{z}, \mathbf{z}' \in \mathbb{Z}_q^{n_z}$, and $\mathbf{w} \in \mathbb{Z}_q^n$, there exists a linear map $\mathsf{R}(\mathbf{z}, \mathbf{z}', \cdot) : \mathsf{mkE}(\mathbf{z}, \alpha) + \mathsf{mE}(\mathbf{z}, \mathbf{w}) \mapsto \mathsf{mkE}(\mathbf{z}', \alpha) + \mathsf{mE}(\mathbf{z}', \mathbf{w})$.

4.3 Construction: LR-CCA2-secure ABKEM Scheme

Here, we will give the construction of an LR-CCA2-secure ABKEM scheme for a predicate P based on our QA-NIZK argument in Section 3 and an LRPE for P.

Construction. Let

$$\hat{\mathcal{L}}^{\mathsf{ABE}}_\rho := \left\{ ([\mathbf{c}]_1, L) \;\middle|\; \exists \mathbf{s} \in \mathbb{Z}_q^k \text{ s.t. } \mathbf{c} = \begin{pmatrix} \mathbf{A} \\ L((\mathbf{A}'_i)_{i\in[1,n]}) \end{pmatrix} \mathbf{s} \right\},$$

where $\rho := \left([\mathbf{A}]_1, ([\mathbf{A}'_i]_1)_{i\in[1,n]}\right) \in \mathbb{G}_1^{(k+1)\times k} \times (\mathbb{G}_1^{(k+1)\times k})^n$, $k \geq 1$ is determined by the underlying assumption, and L is a $\mathbb{Z}_q$-linear map. Let $\Pi = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ be an OT-SS QA-NIZK argument for $\hat{\mathcal{L}} := \{\hat{\mathcal{L}}^{\mathsf{ABE}}_\rho\}_\rho$. Let $(\mathsf{sE}, \mathsf{mE}, \mathsf{mkE}, \mathsf{rE}, \mathsf{rkE}, \mathsf{sD}, \mathsf{rD})$ be an LRPE for P. Our $\mathsf{ABKEM} = (\mathsf{Setup}, \mathsf{KGen}, \mathsf{Encap}, \mathsf{Decap}, \mathsf{UpdateMSK}, \mathsf{UpdateSK})$ is defined in Figure 7.

If we instantiate ABKEM with our QA-NIZK argument $\Pi_{\mathsf{OT\text{-}SS}}$ in Section 3.2 that is secure under the $\mathcal{D}_1$-MDDH assumption, then its ciphertext is only 2 group elements longer than that of the original scheme [11].

Correctness and Security. The correctness of our ABKEM follows readily from the correctness of Zhang et al.'s ABE scheme [11] and the perfect completeness of Π. Next, we

$\underline{\mathsf{Setup}(1^\lambda, \mathcal{X}, \mathcal{Y})}$:
$\mathbf{A}, \mathbf{B} \leftarrow\$\, \mathcal{D}_k, \mathbf{k} \leftarrow\$\, \mathbb{Z}_q^{k+1}, \mathbf{r} \leftarrow\$\, \mathbb{Z}_q^k$
$\mathbf{z} \leftarrow\$\, \mathbb{Z}_q^{n_z}, \mathbf{W}_1, \ldots, \mathbf{W}_n \leftarrow\$\, \mathbb{Z}_q^{(k+1)\times(k+1)}$
$[\mathbf{p}_0]_2 := [\mathbf{Br}]_2 \in \mathbb{G}_2^{k+1}$
$[\mathbf{p}_1]_2 := \mathsf{mE}(\mathbf{z}, ([\mathbf{W}_i\mathbf{Br}]_2)_{i\in[1,n]}) + \mathsf{mkE}(\mathbf{z}, [\mathbf{k}]_2) \in \mathbb{G}_2^{n_m(k+1)}$
$\boxed{\rho := ([\mathbf{A}]_1, ([\mathbf{W}_i^\top\mathbf{A}]_1)_{i\in[1,n]}), \mathsf{crs} \leftarrow\$\, \mathsf{Gen}(\mathsf{par}, \rho)}$
$\mathsf{mpk} := \left([\mathbf{A}]_1, [\mathbf{B}]_2, \left\{[\mathbf{W}_i^\top\mathbf{A}]_1, [\mathbf{W}_i\mathbf{B}]_2\right\}_{i\in[1,n]}, [\mathbf{k}^\top\mathbf{A}]_T, \boxed{\mathsf{crs}}\right)$
$\mathsf{msk} := (\mathbf{z}, [\mathbf{p}_0]_2, [\mathbf{p}_1]_2)$
Return $(\mathsf{mpk}, \mathsf{msk})$

$\underline{\mathsf{KGen}(\mathsf{msk} =: (\mathbf{z}, [\mathbf{p}_0]_2, [\mathbf{p}_1]_2), y)}$:
$\mathbf{r}' \leftarrow\$\, \mathbb{Z}_q^k, \mathbf{z}' \leftarrow\$\, \mathbb{Z}_q^{n_z}$
$[\mathbf{t}_0]_2 := [\mathbf{p}_0]_2 + [\mathbf{Br}']_2 \in \mathbb{G}_2^{k+1}$
$[\mathbf{t}_1]_2 := \mathsf{rE}(y, \mathbf{z}', ([\mathbf{W}_i\mathbf{Br}']_2)_{i\in[1,n]}) + \mathsf{D}(y, \mathsf{R}(\mathbf{z}, \mathbf{z}', [\mathbf{p}_1]_2)) \in \mathbb{G}_2^{n_r(k+1)}$
Return $\mathsf{sk}_y := (\mathbf{z}', [\mathbf{t}_0]_2, [\mathbf{t}_1]_2)$

$\underline{\mathsf{Encap}(\mathsf{mpk}, x)}$:
$\mathbf{s} \leftarrow\$\, \mathbb{Z}_q^k$
$[\mathbf{A}_x]_1 := \left[\begin{array}{c}\mathbf{A}\\ \mathsf{sE}(x, (\mathbf{W}_i^\top\mathbf{A})_{i\in[1,n]})\end{array}\right]_1$
$[\mathbf{c}]_1 := [\mathbf{A}_x\mathbf{s}]_1 \in \mathbb{G}_1^{(n_s+1)(k+1)}$
$\boxed{\pi := \mathsf{Prove}(\mathsf{crs}, ([\mathbf{c}]_1, \mathsf{sE}(x, \cdot)), \mathbf{s})}$
$\mathsf{ct}_x := ([\mathbf{c}]_1, \boxed{\pi}), \mathsf{K} := [\mathbf{k}^\top\mathbf{As}]_T \in \mathbb{G}_T^k =: \mathcal{K}$
Return $(\mathsf{K}, \mathsf{ct}_x)$

$\underline{\mathsf{Decap}((\mathsf{sk}_y =: (\mathbf{z}', [\mathbf{t}_0]_2, [\mathbf{t}_1]_2), y), \mathsf{ct}_x =: ([\mathbf{c}]_1, \boxed{\pi}))}$:
$\boxed{\text{If } \mathsf{Ver}(\mathsf{crs}, ([\mathbf{c}]_1, \mathsf{sE}(x, \cdot)), \pi) = 0\text{: Return } \perp}$
Parse $[\mathbf{c}]_1 =: ([\mathbf{c}_0]_1^\top, [\mathbf{c}_1]_1^\top)^\top \in \mathbb{G}_1^{k+1} \times \mathbb{G}_1^{n_s(k+1)}$
$\mathsf{K} := [\mathbf{c}_0]_1^\top \circ \mathsf{rD}(x, y, \mathbf{z}', [\mathbf{t}_1]_2) - \mathsf{sD}(x, y, \mathbf{z}', [\mathbf{c}_1]_1^\top) \circ [\mathbf{t}_0]_2$
Return K

$\underline{\mathsf{UpdateMSK}(\mathsf{mpk}, \mathsf{msk} := (\mathbf{z}, [\mathbf{p}_0]_2, [\mathbf{p}_1]_2))}$:
$\mathbf{r}' \leftarrow\$\, \mathbb{Z}_q^k, \mathbf{z}' \leftarrow\$\, \mathbb{Z}_q^{n_z}$
$[\mathbf{p}_0']_2 := [\mathbf{p}_0]_2 + [\mathbf{Br}']_2 \in \mathbb{G}_2^{k+1}$
$[\mathbf{p}_1']_2 := \mathsf{mE}(\mathbf{z}', ([\mathbf{W}_i\mathbf{Br}']_2)_{i\in[1,n]}) + \mathsf{R}(\mathbf{z}, \mathbf{z}', [\mathbf{p}_1]_2) \in \mathbb{G}_2^{n_m(k+1)}$
Return $\mathsf{msk}' := (\mathbf{z}', [\mathbf{p}_0']_2, [\mathbf{p}_1']_2)$

$\underline{\mathsf{UpdateSK}(\mathsf{mpk}, \mathsf{sk}_y := (\mathbf{z}, [\mathbf{t}_0]_2, [\mathbf{t}_1]_2), y)}$:
$\mathbf{r}' \leftarrow\$\, \mathbb{Z}_q^k, \mathbf{z}' \leftarrow\$\, \mathbb{Z}_q^{n_z}$
$[\mathbf{t}_0']_2 := [\mathbf{t}_0]_2 + [\mathbf{Br}']_2 \in \mathbb{G}_2^{k+1}$
$[\mathbf{t}_1']_2 := \mathsf{rE}(\mathbf{z}', ([\mathbf{W}_i\mathbf{Br}']_2)_{i\in[1,n]}) + [\mathbf{t}_1]_2 \in \mathbb{G}_2^{n_r(k+1)}$
Return $\mathsf{sk}_y' := (\mathbf{z}', [\mathbf{t}_0']_2, [\mathbf{t}_1']_2)$

Fig. 7 Our LR-CCA2-secure ABKEM scheme ABKEM. The boxed parts are differences from [11]'s LR-CPA-secure ABE scheme.

show the security of our ABKEM.

Theorem 2: If the $\mathcal{D}_k$-MDDH problem in $\mathbb{G}_1$ is hard and Π is an OT-SS QA-NIZK argument, then ABKEM defined in Figure 7 is $(\ell_{\mathsf{msk}}, \ell_{\mathsf{sk}})$-LR-CCA2-secure, where ℓ_{msk} and ℓ_{sk} is derived from the parameters of the underlying LRPE.

Remark 3 (On the concrete instantiations and leakage bounds): Zhang et al. [11] proposed LRPE schemes that correspond to IPE, non-zero IPE, (doubly) spatial encryption, KP/CP-ABE for boolean formulae and arithmetic formulae, and broadcast encryption, and all of them have the same leakage bounds:

$$\ell_{\mathsf{msk}} \le (n_z - 1)\log q + \log(1 - 1/q) + 2 - \omega(\log\lambda),$$
$$\ell_{\mathsf{sk}} \le (n_z - 1)\log q + \log(1 - 1/q) + 2 - \omega(\log\lambda),$$

where $n_z \ge 1$ is an arbitrary integer and k is determined by the underlying assumption. (Larger n_z guarantees higher leakage resilience, but requires longer keys.) By instantiating our construction with their LRPE schemes, we have corresponding LR-CCA2-secure ABE schemes (i.e., IPE, KP/CP-ABE, and so on) with the same leakage bounds.

Proof. Our proof is similar to that of [11, Theorem 1]. Hence, we only sketch our proof here and emphasize the differences.

At a high level, the proof basically follows the dual system methodology [37] and Cramer-Shoup technique [38]. To prove the security, we first give names of various forms of ciphetexts and secret keys that will be used. Let $\mathbf{a}^\perp \in \ker(\mathbf{A})$ and $\mathbf{b}^\perp \in \ker(\mathbf{B})$. A ciphertext under an attribute $x \in \mathcal{X}$ has the following forms:

(Normal): A normal ciphetext is generated as in the actual scheme.

(Semi-Functional): A semi-functional (SF) ciphertext is the same as normal ciphertext except that $\mathbf{As}$ is replaced by $\mathbf{As} + \mathbf{b}^\perp\hat{s}$, where $\hat{s} \leftarrow\$\, \mathbb{Z}_q$. That is,

$$\mathsf{ct}_x = \left([\mathbf{A}_x\mathbf{s}]_1 \boxed{+ \left[\begin{array}{c}\mathbf{b}^\perp\hat{s}\\ \mathsf{sE}(x, (\mathbf{W}_i^\top\mathbf{b}^\perp\hat{s})_{i\in[1,n]})\end{array}\right]_1}, \pi\right).$$

(Invalid): A ciphertext $\mathsf{ct}_x = ([\mathbf{c}]_1, \pi)$ is invalid when $\mathbf{c} \notin \mathsf{Span}(\mathbf{A}_x)$.

A secret key for an attribute $y \in \mathcal{Y}$ can be one of the following forms:

(Normal): A normal secret key is generated as in the actual scheme.

(Pseudo-Normal): A pseudo-normal secret key is the same as normal secret key except that $\mathbf{B}(\mathbf{r} + \mathbf{r}')$ is replaced by $\mathbf{B}(\mathbf{r} + \mathbf{r}') + \mathbf{a}^\perp\hat{r}$, where $\hat{r} \leftarrow\$\, \mathbb{Z}_q$.

(Pseudo-SF): A pseudo-SF secret key is the same as pseudo-normal secret key except that $\mathbf{k}$ is replaced by $\mathbf{k} + \mathbf{a}^\perp\alpha$, where $\alpha \leftarrow\$\, \mathbb{Z}_q$.

(SF): An SF secret key is the same as pseudo-sf secret key except that $\mathbf{B}(\mathbf{r} + \mathbf{r}') + \mathbf{a}^\perp\hat{s}$ is replaced by $\mathbf{B}(\mathbf{r} + \mathbf{r}')$.

Remark 4 (Decapsulation capability of each secret key): We note that the decapsulation results are the same regardless of the form of secret key as long as decapsulated ciphertexts are not invalid. Furthermore, the results are always in the form of $[\mathbf{k}^\top\mathbf{As}]_T$ for some $\mathbf{s} \in \mathbb{Z}_q^k$, and hence they completely hide α used in pseudo-SF and SF secret keys.

We will show that for any adversary $\mathcal{A}$ that makes at most Q queries to REVEAL and LEAK, there exist adversaries $\mathcal{B}$, $\mathcal{B}'$, and $\mathcal{B}''$ with

$$\mathsf{Adv}^{\text{LR-CCA}}_{\mathsf{ABKEM},\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{MDDH}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) + 2Q\mathsf{Adv}^{\mathsf{MDDH}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}'}(\lambda)$$

$$+ \mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}''}(\lambda) + Q2^{-\omega(\log\lambda)}. \quad (6)$$

We define the following sequence of games to prove the security.

- G_0: This is the real security game defined in Figure 6.
- G_1: This game is the same as G_0 except that $([\mathbf{c}^*]_1, \pi^*, \mathsf{K}^*_0)$ outputted by CHAL are computed as follows:

$$[\mathbf{c}^*]_1 = \left[\begin{pmatrix}\mathbf{I}_{k+1}\\ \mathsf{sE}(x, (\mathbf{W}_i^\top)_{i\in[1,n]})\end{pmatrix}\mathbf{c}^*_0\right]_1,$$
$$\pi^* = \mathsf{Sim}(\mathsf{crs}, \mathsf{td}, ([\mathbf{c}^*]_1, \mathsf{sE}(x,\cdot))),$$
$$\mathsf{K}^*_0 = [\mathbf{k}^\top\mathbf{c}^*_0]_T,$$

  where $\mathbf{c}^*_0 := \mathbf{As}$.
- G_2: This game is the same as G_1 except that the challenge ciphertext becomes SF. Namely, $\mathbf{c}^*_0 = \mathbf{As}$ is replaced by $\mathbf{c}^*_0 = \mathbf{As} + \mathbf{b}^\perp\hat{s}$.
- G_3: This game is the same as G_2 except that DECAP returns $\perp$ for $\mathcal{A}$'s queries $(\cdot, \mathsf{ct}_x)$ such that ct_x is invalid.
- $\mathsf{G}_{4,i,1}$: This game is the same as $\mathsf{G}_{4,i-1,3}$ except that the i-th keys revealed (or leaked) to $\mathcal{A}$ become pseudo-normal. The game is defined for $i = 1, \ldots, Q$.
- $\mathsf{G}_{4,i,2}$: This game is the same as $\mathsf{G}_{4,i,1}$ except that the i-th key revealed (or leaked) becomes pseudo-SF. The game is defined for $i = 1, \ldots, Q$.
- $\mathsf{G}_{4,i,3}$: This game is the same as $\mathsf{G}_{4,i,2}$ except that the i-th key revealed (or leaked) becomes SF. The game is defined for $i = 1, \ldots, Q$. We set $\mathsf{G}_{4,0,3} := \mathsf{G}_3$.
- G_5: This game is the same as $\mathsf{G}_{4,Q,3}$ except that $\mathsf{K}^*_0 \leftarrow\!\!\$\, \mathcal{K}$.

To prove the security of our scheme, we make some fine-tuning of the game. The LR-CCA2 game does not really generate a secret key and only returns a handle to $\mathcal{A}$ when $\mathcal{A}$ queries to CREATE. Alternatively, the game generates a secret key if the secret key has not generated when $\mathcal{A}$ queries REVEAL, LEAK, or DECAP, and then adds the key to the set Q_T. We note that this makes no difference in $\mathcal{A}$'s view.

In G_5, the view of $\mathcal{A}$ is statistically independent of the challenge bit b. Hence, we have

$$\Pr[\mathsf{G}^{\mathcal{A}}_5 \Rightarrow 1] = 1/2. \quad (7)$$

We complete the proof by establishing the following sequence of lemmas. We omit the proof of Lemmas 8 and 11 to 14 as they are the same as those of Lemmas 2 to 6 in [11, Section 5.2].

Lemma 7 (G_0 to G_1): $\Pr[\mathsf{LR\text{-}CCA}^{\mathcal{A}}_{\mathsf{CML}} \Rightarrow 1] = \Pr[\mathsf{G}^{\mathcal{A}}_0 \Rightarrow 1] = \Pr[\mathsf{G}^{\mathcal{A}}_1 \Rightarrow 1]$.

Proof. G_0 is the real security game. In G_1, the change in the way generating $[\mathbf{c}_1]^*$ and K^*_0 is conceptual since

$$\mathbf{c}^* = \begin{pmatrix}\mathbf{A}\\ \mathsf{sE}(x, (\mathbf{W}_i^\top\mathbf{A})_{i\in[1,n]})\end{pmatrix}\mathbf{s}$$
$$= \begin{pmatrix}\mathbf{I}_{k+1}\\ \mathsf{sE}(x, (\mathbf{W}_i^\top)_{i\in[1,n]})\end{pmatrix}\mathbf{As}$$
$$= \begin{pmatrix}\mathbf{I}_{k+1}\\ \mathsf{sE}(x, (\mathbf{W}_i^\top)_{i\in[1,n]})\end{pmatrix}\mathbf{c}^*_0,$$
$$\mathsf{K}^*_0 = \mathbf{k}^\top\mathbf{As} = \mathbf{k}^\top\mathbf{c}^*_0.$$

Moreover, we simulate the QA-NIZK proof π^* in $\mathsf{CHAL}(x^*)$ by using Π's zero-knowledge simulator. By the perfect zero-knowledge property of Π, G_1 is identical to G_0. □

Lemma 8 (G_1 to G_2): There is an adversary $\mathcal{B}$ breaking the $\mathcal{D}_k$-MDDH assumption in $\mathbb{G}_1$ with $\mathsf{Adv}^{\mathsf{MDDH}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) \geq |\Pr[\mathsf{G}^{\mathcal{A}}_1 \Rightarrow 1] - \Pr[\mathsf{G}^{\mathcal{A}}_2 \Rightarrow 1]|$.

Lemma 9 (G_2 to G_3): There is an adversary $\mathcal{B}'$ breaking the $\mathsf{OT\text{-}SS}$ of Π with $\mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}'}(\lambda) \geq |\Pr[\mathsf{G}^{\mathcal{A}}_2 \Rightarrow 1] - \Pr[\mathsf{G}^{\mathcal{A}}_3 \Rightarrow 1]|$.

Proof. The difference between G_2 and G_3 happens when $\mathcal{A}$ queries DECAP with $(x, \mathsf{ct}_x = ([\mathbf{c}]_1, \pi))$ such that $\mathbf{c} \notin \mathsf{Span}(\mathbf{A}_{\mathsf{ID}})$ and $\mathsf{Ver}(\mathsf{crs}, ([\mathbf{c}]_1, \mathsf{ID}), \pi) = 1$. The probability of this happening is bounded by $\mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}'}(\lambda)$, and then we have $|\Pr[\mathsf{G}^{\mathcal{A}}_2 \Rightarrow 1] - \Pr[\mathsf{G}^{\mathcal{A}}_3 \Rightarrow 1]| \leq \mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}'}(\lambda)$. □

In the subsequent games, while REVEAL and LEAK use the various types of secret key, DECAP always uses a normal secret key to return the decapsulation result. From Remark 4, this unfairness does not affect $\mathcal{A}$'s view, because the DECAP rejects all invalid ciphertexts by the change in G_3. Furthermore, the DECAP dose not provide additional information to $\mathcal{A}$ since the DECAP returns $\perp$ for $\mathcal{A}$'s queries $(\cdot, ([\mathbf{c}]_1, \pi))$ such that $\mathbf{c} \notin \mathsf{Span}(\mathbf{A}_x)$.

Lemma 10 (G_3 to $\mathsf{G}_{4,0,3}$): $\Pr[\mathsf{G}^{\mathcal{A}}_3 \Rightarrow 1] = \Pr[\mathsf{G}^{\mathcal{A}}_{4,0,3} \Rightarrow 1]$.

Lemma 11 ($\mathsf{G}_{4,i-1,3}$ to $\mathsf{G}_{4,i,1}$): There is an adversary $\mathcal{B}''$ such that $\mathsf{Adv}^{\mathsf{MDDH}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}''}(\lambda) \geq |\Pr[\mathsf{G}^{\mathcal{A}}_{4,i-1,3} \Rightarrow 1] - \Pr[\mathsf{G}^{\mathcal{A}}_{4,i,1} \Rightarrow 1]|$.

Lemma 12 ($\mathsf{G}_{4,i,1}$ to $\mathsf{G}_{4,i,2}$): We have $|\Pr[\mathsf{G}^{\mathcal{A}}_{4,i,1} \Rightarrow 1] - \Pr[\mathsf{G}^{\mathcal{A}}_{4,i,2} \Rightarrow 1]| \leq 2^{-\omega(\log\lambda)}$, as long as the leakage amount of msk and sk are at most ℓ_{msk} and ℓ_{sk} bits, respectively. Here, ℓ_{msk} and ℓ_{sk} is derived from the parameters of the underlying LRPE.

Lemma 13 ($\mathsf{G}_{4,i,2}$ to $\mathsf{G}_{4,i,3}$): There is an adversary $\mathcal{B}''$ such that $\mathsf{Adv}^{\mathsf{MDDH}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}''}(\lambda) \geq |\Pr[\mathsf{G}^{\mathcal{A}}_{4,i,2} \Rightarrow 1] - \Pr[\mathsf{G}^{\mathcal{A}}_{4,i,3} \Rightarrow 1]|$.

Lemma 14 ($\mathsf{G}_{4,Q,3}$ to G_5): $\Pr[\mathsf{G}^{\mathcal{A}}_{4,Q,3} \Rightarrow 1] = \Pr[\mathsf{G}^{\mathcal{A}}_5 \Rightarrow 1]$.

From Lemmas 7 to 14 and Eq. (7), we obtain Eq. (6). □

As a result, we obtain the following corollary when we use our OT-SS QA-NIZK argument in Section 3.2.

Corollary 1: Let $k, k' \geq 1$. If the $\mathcal{D}_k$-MDDH problem in $\mathbb{G}_1$ and the $\mathcal{D}_{k'}$-KerMDH problem in $\mathbb{G}_2$ are hard and $\mathcal{H}$ is a CR hash function family, then our ABKEM is $(\ell_{\mathsf{msk}}, \ell_{\mathsf{sk}})$-LR-CCA2 secure, where ℓ_{msk} and ℓ_{sk} is derived from the parameters of the underlying LRPE.

5. LR-CCA2-secure Identity-Based Encryption with Optimal Leakage Rate

In this section, we show our efficient LR-CCA2-secure identity-based key encapsulation (IBKEM)[†] scheme that is resilient to the leakage of $(1 - o(1))$-fraction of its secret key. Our LR-CCA2-secure IBKEM scheme is based on the LR-CPA-secure IBE scheme by Kurosawa and Phong [9] and our simulation-sound QA-NIZK argument in Section 3.

In Section 5.1, we first give the definition of IBKEM and its security model. In Section 5.2, we then provide our IBKEM scheme, that is secure against *selective*-identity attacks. In Section 5.3, we briefly explain how to extend our scheme to be an adaptive secure variant.

5.1 Definition

IBKEM is a special case of ABKEM where a predicate P is the equality checking predicate, i.e., $P(\mathsf{ID}, \mathsf{ID}') = 1$ if and only if $\mathsf{ID} = \mathsf{ID}'$. Therefore, its syntax and correctness are the same as those of ABKEM shown in Section 4.1.

As the leakage model, we consider the bounded memory leakage (BML) model in this section. There is no time period in this model, thus we do not use key update algorithms. Furthermore, we do not consider the leakage of its master secret key.

Following [13], we define the LR-CCA2-security of an IBKEM in the BML model.

LR-CCA2-Security of IBKEM in the BML model. Let $\ell_{\mathsf{sk}} = \ell_{\mathsf{sk}}(\lambda)$ be a leakage bound for secret keys. An IBKEM scheme IBKEM is ℓ_{sk}-LR-CCA2-secure if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{LR\text{-}CCA}}_{\mathsf{IBKEM},\mathcal{A}}(\lambda) := \left|\Pr[\mathsf{LR\text{-}CCA}^{\mathcal{A}}_{\mathsf{BML}} \Rightarrow 1] - 1/2\right|$ is negligible, where Game $\mathsf{LR\text{-}CCA}_{\mathsf{BML}}$ is defined as in Figure 8.

We say that IBKEM is secure against *selective*-identity attacks (s-LR-CCA2), when the adversary chooses the challenge identity ID^* before seeing any parameters. We denote the corresponding advantage function by $\mathsf{Adv}^{\mathsf{s\text{-}LR\text{-}CCA}}_{\mathsf{IBKEM},\mathcal{A}}(\lambda)$.

We define leakage rate γ of the scheme to be value of $\ell_{\mathsf{sk}}/|\mathsf{sk}_{\mathsf{ID}}|$. We say that the leakage rate of the scheme is *optimal* if $\gamma = 1 - o(1)$.

5.2 Construction: s-LR-CCA2-secure IBKEM Scheme

Here, we give our construction of s-LR-CCA2-secure IBKEM scheme with optimal leakage rate.

Construction. Let $\mathcal{I} := \mathbb{Z}_q$ be an identity space. Let

$$\mathcal{L}^{\mathsf{IBE}}_{\rho} := \left\{ ([\mathbf{c}]_1, \mathsf{ID}) \in \mathbb{G}_1^{2k+\mu} \times \mathbb{Z}_q \;\middle|\; \exists \mathbf{s} \in \mathbb{Z}_q^k \text{ s.t.} \right.$$

$$\left. \mathbf{c} = \begin{pmatrix} \mathbf{A} \\ \mathbf{B} + \mathsf{ID} \cdot \mathbf{B}' \end{pmatrix} \mathbf{s} \right\},$$

where $\rho := ([\mathbf{A}]_1, [\mathbf{B}]_1, [\mathbf{B}']_1) \in \mathbb{G}_1^{(k+\mu)\times k} \times (\mathbb{G}_1^{k\times k})^2$, $k \geq 2$ is determined by the underlying assumption, and $\mu \geq 1$ is an arbitrary integer. Note that $\mathcal{L}^{\mathsf{IBE}}_{\rho}$ is a special case of $\mathcal{L}^{\mathsf{GTLS}}_{\rho}$ where $m = 2$ and x_1 is fixed to 1. As μ increases, the efficiency of the scheme becomes less efficient, but the leakage rate of the scheme becomes greater. Let $\Pi = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ be an OT-SS QA-NIZK argument for $\mathcal{L} := \{\mathcal{L}^{\mathsf{IBE}}_{\rho}\}_{\rho}$. Our $\mathsf{IBKEM} = (\mathsf{Setup}, \mathsf{KGen}, \mathsf{Encap}, \mathsf{Decap})$ is defined in Figure 9.

Similar to our ABKEM, a ciphertext of our IBKEM is only 2 group elements longer than that of the original scheme [9] if we instantiate our IBKEM with our QA-NIZK argument $\Pi_{\mathsf{OT\text{-}SS}}$ that is secure under the $\mathcal{D}_1$-MDDH assumption.

Correctness and Security. The correctness of our IBKEM follows readily from the correctness of Kurosawa-Phong IBE scheme [9] and the perfect completeness of Π. Next, we give the security of our IBKEM.

Theorem 3: If $\mathcal{D}_{k+\mu,k}$-exMDDH problem in $\mathbb{G}_1$ is hard, Π is an OT-SS QA-NIZK argument, and

$$\ell_{\mathsf{sk}} \leq (\mu + k - 1)\log q - \omega(\log \lambda), \tag{8}$$

then IBKEM defined in Figure 9 is ℓ_{sk}-s-LR-CCA2-secure.

Remark 5 (Leakage rate of our scheme): From Eq. (8), we have

$$\gamma = \frac{\ell_{\mathsf{sk}}}{|\mathsf{sk}_{\mathsf{ID}}|} = \frac{(\mu + k - 1)\log q - \omega(\log \lambda)}{(\mu + 2k)\log q} = 1 - \frac{(k+1)\log q + \omega(\log \lambda)}{(\mu + 2k)\log q}.$$

[†]Similarly to Section 4, we only focus on IBKEMs since we can convert an LR-CCA2-secure IBKEM scheme to an LR-CCA2-secure IBE scheme using a symmetric encryption scheme. For details, we refer the reader to [36].

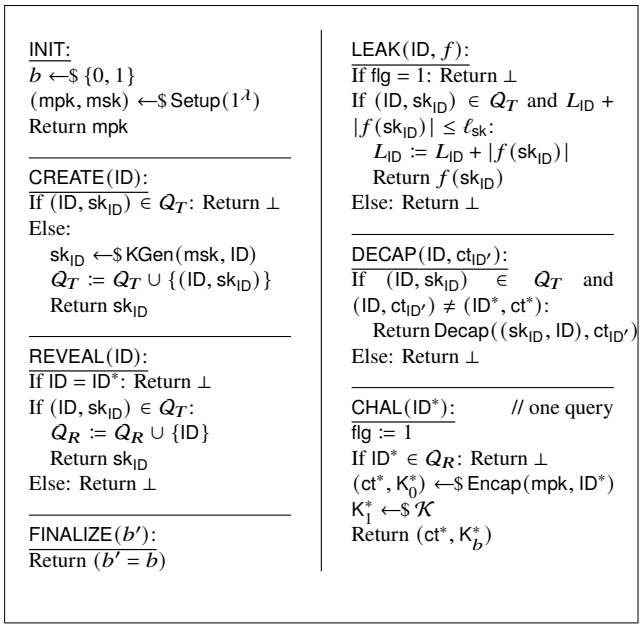


Fig. 8 Security game $\mathsf{LR\text{-}CCA}_{\mathsf{BML}}$.

$\underline{\mathsf{Setup}(1^\lambda)}$:
$\mathbf{A} \leftarrow\$ \mathcal{D}_{k+\mu,k}, \mathbf{B} \leftarrow\$ \mathbb{Z}_q^{k\times k}, \mathbf{k} \leftarrow\$ \mathbb{Z}_q^k,$
$\boxed{\rho := ([\mathbf{A}]_1, [\mathbf{B}]_1, [\mathbf{I}_k]_1), (\mathsf{crs}, \mathsf{td}) \leftarrow\$ \mathsf{Gen}(\mathsf{par}, \rho)}$
$\mathsf{mpk} := ([\mathbf{A}]_1, [\mathbf{B}]_1, [\mathbf{k}]_1, \boxed{\mathsf{crs}}), \mathsf{msk} := (\mathbf{A}, \mathbf{B}, \mathbf{k})$
Return $(\mathsf{mpk}, \mathsf{msk})$

$\underline{\mathsf{KGen}(\mathsf{msk} =: (\mathbf{A}, \mathbf{B}, \mathbf{k}), \mathsf{ID})}$:
$\mathbf{t}_1 \leftarrow\$ \mathbb{Z}_q^{\mu+k}, \mathbf{t}_0 := \overline{\mathbf{A}}^{-1^\top}\left(\mathbf{k} - \left(\mathbf{B} + \mathsf{ID}\cdot\mathbf{I}_k\right)^\top \mathbf{t}_1\right) \in \mathbb{Z}_q^k$
Return $\mathsf{sk}_{\mathsf{ID}} := \begin{bmatrix}\mathbf{t}_0\\ \mathbf{t}_1\end{bmatrix}_2 \in \mathbb{G}_2^{2k+\mu}$

$\underline{\mathsf{Encap}(\mathsf{mpk}, \mathsf{ID})}$:
$\mathbf{s} \leftarrow\$ \mathbb{Z}_q^k$
$[\mathbf{A}_{\mathsf{ID}}]_1 := \begin{bmatrix}\mathbf{A}\\ \mathbf{B} + \mathsf{ID}\cdot\mathbf{I}_k\end{bmatrix}_1 \in \mathbb{G}_1^{(2k+\mu)\times k}, [\mathbf{c}]_1 := [\mathbf{A}_{\mathsf{ID}}\mathbf{s}]_1 \in \mathbb{G}_1^{2k+\mu}$
$\boxed{\pi := \mathsf{Prove}(\mathsf{crs}, ([\mathbf{c}]_1, \mathsf{ID}), \mathbf{s})}$
$\mathsf{ct}_{\mathsf{ID}} := ([\mathbf{c}]_1, \boxed{\pi}), \mathsf{K} := [\mathbf{k}^\top\mathbf{s}]_T \in \mathbb{G}_T =: \mathcal{K}$
Return $(\mathsf{K}, \mathsf{ct}_{\mathsf{ID}})$

$\underline{\mathsf{Decap}((\mathsf{sk}_{\mathsf{ID}} =: [\mathbf{t}]_2, \mathsf{ID}), \mathsf{ct}_{\mathsf{ID}'} =: ([\mathbf{c}]_1, \boxed{\pi}))}$:
$\boxed{\text{If } \mathsf{Ver}(\mathsf{crs}, ([\mathbf{c}]_1, \mathsf{ID}), \pi) = 0\text{: Return } \perp}$
Return $\mathsf{K} := [\mathbf{c}]_1^\top \circ [\mathbf{t}]_2$

Fig. 9 Our s-LR-CCA2-secure IBKEM scheme IBKEM. The boxed parts are differences from [9]'s scheme.

If $\mu = \omega(\log\lambda)$, with k fixed, the leakage rate γ achieves $1 - o(1)$ and then our scheme has optimal leakage rate because $\log q = O(\lambda)$.

We give a simple example of a parameter setting to obtain the rate we want. We assume that $k = 2$ and $\log q = \lambda$, and we use 2λ as $\omega(\log\lambda)$ for λ-bit security. If we want to set the rate to $3/4$, then we should set $\mu = 16$ since we obtain the rate $\gamma = 1 - (3+2)/(16+4) = 3/4$ as desired.

Remark 6 (On the efficiency of the scheme): To achieve the optimal leakage rate, our LR-CCA2-secure IBE scheme requires $\omega(\log\lambda)$ group elements in the ciphertext. Such a ciphertext overhead is currently unavoidable even for LR-*CPA*-secure *PKE* schemes that achieve the optimal leakage rate (e.g., [4]). Furthermore, the ciphertext size of our LR-CCA2-secure scheme is almost the same as the state-of-the-art LR-CPA-secure IBE scheme [9]. From these facts, our scheme is (currently) efficient.

Proof. We will show that for any adversary $\mathcal{A}$, there exist adversaries $\mathcal{B}$ and $\mathcal{B}'$ with

$$\mathsf{Adv}^{\text{s-LR-CCA}}_{\mathsf{IBKEM},\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{exMDDH}}_{\mathbb{G}_1,\mathcal{D}_{k+\mu,k},\mathcal{B}}(\lambda) + \mathsf{Adv}^{\text{OT-SS}}_{\Pi,\mathcal{B}'}(\lambda) + 1/q + 2^{-\omega(\log\lambda)}. \quad (9)$$

We define the following sequence of games to prove the security.

- G_0: This game is the real security game defined in Figure 8.
- G_1: This game is the same as G_0 except that π^* in the challenge ciphertext is computed via $\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, ([\mathbf{c}^*]_1, \mathsf{ID}^*))$ instead of $\mathsf{Prove}(\mathsf{crs}, ([\mathbf{c}^*]_1, \mathsf{ID}^*), \mathbf{s})$.
- G_2: This game is the same as G_1 except for the following changes:
  - $[\mathbf{B}]_1$ and $[\mathbf{k}]_1$ in the master public key are computed as follows:
    $$[\mathbf{B}]_1 := [\mathbf{R}^*\mathbf{A} - \mathsf{ID}^*\cdot\mathbf{I}_k]_1 \text{ and}$$
    $$[\mathbf{k}]_1 := \left[\begin{pmatrix}\mathbf{A}\\ \mathbf{R}^*\mathbf{A}\end{pmatrix}^\top \mathbf{t}^*\right]_1,$$
    where $\mathbf{R}^* \leftarrow\$ \mathbb{Z}_q^{k\times(k+\mu)}$ and $\mathbf{t}^* \leftarrow\$ \mathbb{Z}_q^{2k+\mu}$. In the following games, the game s-LR-CCA2 sets $[\mathbf{t}^*]_2$ as a secret key for the challenge identity ID^*. Then, we have
    $$\mathbf{A}_{\mathsf{ID}} = \begin{pmatrix}\mathbf{A}\\ \mathbf{B} + \mathsf{ID}\cdot\mathbf{I}_k\end{pmatrix} = \begin{pmatrix}\mathbf{A}\\ \mathbf{R}^*\mathbf{A} + (\mathsf{ID} - \mathsf{ID}^*)\cdot\mathbf{I}_k\end{pmatrix}.$$
  - K_0^* outputted by CHAL is computed by $[\mathbf{c}^{*\top}\mathbf{t}^*]_T$.
  - $\mathsf{sk}_{\mathsf{ID}}$ for $\mathsf{ID} \neq \mathsf{ID}^*$ is generated as follows:
    $$\mathbf{t}_0' \leftarrow\$ \mathbb{Z}_q^{k+\mu},\ [\mathbf{t}_1']_2 := \left[\frac{-\mathbf{A}^\top\mathbf{t}_0' + \mathbf{k}}{\mathsf{ID} - \mathsf{ID}^*}\right]_2,$$
    $$\mathsf{sk}_{\mathsf{ID}} = [\mathbf{t}]_2 := \begin{bmatrix}\mathbf{t}_0' - \mathbf{R}^{*\top}\mathbf{t}_1'\\ \mathbf{t}_1'\end{bmatrix}_2.$$
    We note that these values can be generated without knowing $\mathbf{A}$, while with knowing $[\mathbf{A}]_1$ and $[\mathbf{A}]_2$.
- G_3: This game is the same as G_2 except that $[\mathbf{c}^*]_1$ in the challenge ciphertext is sampled from $\mathbb{G}_1^{2k+\mu}$ uniformly at random.
- G_4: This game is the same as G_3 except that DECAP returns $\perp$ for $\mathcal{A}$'s queries $(\mathsf{ID}, \mathsf{ct} = ([\mathbf{c}]_1, \pi))$ such that $\mathbf{c} \notin \mathsf{Span}(\mathbf{A}_{\mathsf{ID}})$.
- G_5: This game is the same as G_4 except that K_0^* outputted by CHAL is sampled from $\mathbb{G}_T$ uniformly at random.

In G_5, the view of the adversary is statistically independent of the challenge bit b. Hence, we have

$$\Pr[\mathsf{G}_5^{\mathcal{A}} \Rightarrow 1] = 1/2. \quad (10)$$

To complete the proof, we prove the following sequence of lemmas.

Lemma 15 (G_0 to G_1): $\Pr[\text{LR-CCA}^{\mathcal{A}}_{\mathsf{BML}} \Rightarrow 1] = \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]$.

Proof. G_0 is the real security game. In G_1, we simulate the QA-NIZK proof π^* in $\mathsf{CHAL}(\mathsf{ID}^*)$ by using Π's zero-knowledge simulator. By the perfect zero-knowledge property of Π, we have $\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]$. □

Lemma 16 (G_1 to G_2): $\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]$.

Proof. The distributions of all values is not affected by the

changes in G_2. Then, we have $\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]$. □

Lemma 17 (G_2 to G_3): There is an adversary $\mathcal{B}$ that solves the $\mathcal{D}_{k+\mu,k}$-exMDDH problem in $\mathbb{G}_1$ with $\mathsf{Adv}^{\mathsf{exMDDH}}_{\mathbb{G}_1,\mathcal{D}_{k+\mu,k},\mathcal{B}}(\lambda) \geq |\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1]| - 1/q$.

Proof. Using $\mathcal{A}$, we can construct $\mathcal{B}$ that solves the $\mathcal{D}_{k+\mu,k}$-exMDDH problem in $\mathbb{G}_1$ as follows. Let $(\mathcal{G}, [\mathbf{A}]_1, [\mathbf{A}]_2, [\mathbf{u}]_1)$ be an instance of $\mathcal{D}_{k+\mu,k}$-exMDDH problem, where either $\mathbf{u} = \mathbf{A}\mathbf{w}$ for $\mathbf{w} \leftarrow\!\$\, \mathbb{Z}_q^k$ or $\mathbf{u} \leftarrow\!\$\, \mathbb{Z}_q^{k+1}$, and let ID^* be a challenge identity chosen by $\mathcal{A}$. Clearly, $\mathcal{B}$ can generate mpk and $\mathsf{sk}_{\mathsf{ID}}$ using $[\mathbf{A}]_1$ and $[\mathbf{A}]_2$, respectively. Hence, $\mathcal{B}$ can simulate INIT, REVEAL, LEAK, and DECAP. We only focus on simulating CHAL by $\mathcal{B}$.

When $\mathcal{A}$ queries to CHAL, $\mathcal{B}$ simulates the values outputted by CHAL as follows:

$$[\mathbf{c}^*]_1 := \begin{bmatrix} \mathbf{u} \\ \mathbf{R}^*\mathbf{u} \end{bmatrix}_1,\ \pi := \mathsf{Sim}(\mathsf{crs}, \mathsf{td}, ([\mathbf{c}^*]_1, \mathsf{ID}^*)),$$
$$\mathsf{K}_0^* := [\mathbf{c}^{*\top}\mathbf{t}^*]_T,\ \mathsf{K}_1^* \leftarrow\!\$\, \mathbb{G}_T.$$

Then, $\mathcal{B}$ sends $([\mathbf{c}^*]_1, \pi, \mathsf{K}_b^*)$ to $\mathcal{A}$. To conclude this proof, we show that the distribution of $\mathbf{c}^*$ is the same as one in G_2 if $\mathbf{u} = \mathbf{A}\mathbf{w}$ and in G_3 otherwise.

- Case $\mathbf{u} = \mathbf{A}\mathbf{w}$: In this case, we have
$$\mathbf{c}^* := \begin{pmatrix} \mathbf{u} \\ \mathbf{R}^*\mathbf{u} \end{pmatrix} = \begin{pmatrix} \mathbf{A} \\ \mathbf{R}^*\mathbf{A} \end{pmatrix} \mathbf{w} = \mathbf{A}_{\mathsf{ID}^*}\mathbf{w}.$$
Hence, the distribution of $\mathbf{c}^*$ is the same as one in G_2.
- Case $\mathbf{u} \leftarrow\!\$\, \mathbb{Z}_q^{k+\mu}$: It is sufficient to show that $\mathbf{R}^*\mathbf{u}$ is uniformly distributed over $\mathbb{Z}_q^k$ from $\mathcal{A}$'s view point. With probability $1 - 1/q$, $\mathbf{u}$ is linearly independent of $\mathbf{A}$ since $\mathbf{A}$ is full rank and $\mathbf{u}$ is uniformly distributed over $\mathbb{Z}_q^{k+\mu}$. Hence, $\mathbf{R}^*\mathbf{u}$ is uniformly distributed over $\mathbb{Z}_q^k$ even given $\mathbf{A}$, $\mathbf{R}^*\mathbf{A}$, and $\mathbf{u}$, which are all the information that $\mathcal{A}$ knows. Therefore, the distribution of $\mathbf{c}^*$ is the same as one in G_3.

From these, we have $|\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1]| \leq \mathsf{Adv}^{\mathsf{exMDDH}}_{\mathbb{G}_1,\mathcal{D}_{k+\mu,k},\mathcal{B}}(\lambda) + 1/q$. □

Lemma 18 (G_3 to G_4): There is an adversary $\mathcal{B}'$ breaking the OT-SS of Π with $\mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}'}(\lambda) \geq |\Pr[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1]|$.

Proof. The difference between G_3 and G_4 happens when an adversary queries DECAP with $(\mathsf{ID}, \mathsf{ct} = ([\mathbf{c}]_1, \pi))$ such that $\mathbf{c} \notin \mathsf{Span}(\mathbf{A}_{\mathsf{ID}})$ and $\mathsf{Ver}(\mathsf{crs}, ([\mathbf{c}]_1, \mathsf{ID}), \pi) = 1$. The probability of this happening is bounded by $\mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}'}(\lambda)$, and then we have $|\Pr[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1]| \leq \mathsf{Adv}^{\mathsf{OT\text{-}SS}}_{\Pi,\mathcal{B}'}(\lambda)$. □

By the change in G_4, in the following game, $\mathcal{A}$ learns information about $\mathbf{t}^*$ only from $\mathbf{k}$ in mpk and $\{f(\mathsf{sk}_{\mathsf{ID}^*})\}$ which is ℓ_{sk}-bit leakage of $\mathsf{sk}_{\mathsf{ID}}$ since DECAP returns $\perp$ for $\mathcal{A}$'s queries $(\mathsf{ID}, ([\mathbf{c}]_1, \pi))$ such that $\mathbf{c} \notin \mathsf{Span}(\mathbf{A}_{\mathsf{ID}})$.

Lemma 19 (G_4 to G_5): $|\Pr[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_5^{\mathcal{A}} \Rightarrow 1]| \leq 2^{-\omega(\log\lambda)}$.

To prove this lemma, we use the following lemma.

Lemma 20 (Generalized leftover hash lemma [41]): Let $\mathcal{H} = \{\mathsf{H} : \mathcal{X} \to \mathcal{Y}\}$ be a universal hash function family, that is $\Pr[\mathsf{H}(x) = \mathsf{H}(x') \mid \mathsf{H} \leftarrow\!\$\, \mathcal{H}] = 1/|\mathcal{Y}|$ for any $x \neq x' \in \mathcal{X}$. Let $f : \mathcal{X} \to \mathcal{Z}$ be any function, where $\mathcal{Z}$ is a finite set. Then, for any random variable X over $\mathcal{X}$, we have

$$\Delta((\mathsf{H}, \mathsf{H}(X), f(X)), (\mathsf{H}, U_{\mathcal{Y}}, f(X))) \leq \frac{1}{2}\sqrt{\gamma(X) \cdot |\mathcal{Y}| \cdot |\mathcal{Z}|},$$

where $\Delta(\cdot,\cdot)$ denotes the statistical distance between two distributions, $\mathsf{H} \leftarrow\!\$\, \mathcal{H}$, $U_{\mathcal{Y}}$ is the uniform distribution over $\mathcal{Y}$, and $\gamma(X) := \max_x \Pr[X = x]$.

Proof of Lemma 19. To bound this, we show that the two distributions $(\mathbf{c}^*, \mathbf{c}^{*\top}\mathbf{t}^*, \mathbf{k}, \{f(\mathsf{sk}_{\mathsf{ID}^*})\})$ and $(\mathbf{c}^*, u, \mathbf{k}, \{f(\mathsf{sk}_{\mathsf{ID}^*})\})$ are statistically indistinguishable, where $u \leftarrow\!\$\, \mathbb{Z}_q$. We can see that $\mathsf{H}_{\mathbf{c}^*}(\mathbf{t}^*) := \mathbf{c}^{*\top}\mathbf{t}^*$ is a universal hash function. By Lemma 20 and Eq. (8), we have

$$\begin{aligned}
&\Delta\left((\mathbf{c}^*, \mathbf{c}^{*\top}\mathbf{t}^*, \mathbf{k}, \{f(\mathsf{sk}_{\mathsf{ID}^*})\}), (\mathbf{c}^*, u, \mathbf{k}, \{f(\mathsf{sk}_{\mathsf{ID}^*})\})\right) \\
&\leq \frac{1}{2}\sqrt{q^{-(2k+\mu)} \cdot q \cdot (q^k 2^{\ell_{\mathsf{sk}}})} \\
&= \frac{1}{2}\sqrt{2^{-(k+\mu-1)\log q + \ell_{\mathsf{sk}}}} \\
&= 2^{-\omega(\log\lambda)}.
\end{aligned}$$

Hence, we have $|\Pr[\mathsf{G}_4^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_5^{\mathcal{A}} \Rightarrow 1]| \leq 2^{-\omega(\log\lambda)}$. □

From Lemmas 15 to 19 and Eq. (10), we obtain Eq. (9). □

As a result, we obtain the following corollary when we use our OT-SS QA-NIZK argument in Section 3.2.

Corollary 2: Let $k \geq 2$ and $k' \geq 1$. If the $\mathcal{D}_{k+\mu,k}$-exMDDH problem in $\mathbb{G}_1$ and the $\mathcal{D}_{k'}$-KerMDH problem in $\mathbb{G}_2$ are hard, $\mathcal{H}$ is a CR hash function family, and Eq. (8) holds, then our IBKEM is ℓ_{sk}-s-LR-CCA2-secure.

5.3 Adaptive Security Variant

Here, we briefly explain how to extend our scheme to be an adaptive secure variant. We use the same techniques of [9], [19], [20].

Let $m \in \mathbb{N}$, and let $\mathcal{I} := \{0,1\}^m$ be an identity space. Our LR-CCA2-secure IBKEM scheme is obtained by changing the scheme in previous section as follows:

- We set $\mathsf{mpk} := ([\mathbf{A}]_1, \boxed{\{[\mathbf{B}_i]_1\}_{i\in[0,m]}}, [\mathbf{k}]_1, \mathsf{crs})$ and $\mathsf{msk} := (\mathbf{A}, \boxed{\{\mathbf{B}_i\}_{i\in[0,m]}}, \mathbf{k})$, where $\mathbf{B}_0, \ldots, \mathbf{B}_m \in \mathbb{Z}_q^{k\times k}$.
- For $\mathsf{ID} := (\mathsf{ID}_1, \ldots, \mathsf{ID}_m) \in \{0,1\}^m$, we set

$$\mathbf{A}_{\mathsf{ID}} := \begin{pmatrix} \mathbf{A} \\ \boxed{\mathbf{B}_0 + \sum_{i=1}^{m} \mathsf{ID}_i \cdot \mathbf{B}_i} \end{pmatrix} \in \mathbb{Z}_q^{(2k+\mu)\times k}.$$

- We use a QA-NIZK argument for a language

$$\mathcal{L}_\rho^{\mathsf{IBE}'} := \left\{ ([\mathbf{c}]_1, \mathsf{ID}) \mid \exists \mathbf{s} \in \mathbb{Z}_q^k \text{ s.t. } \mathbf{c} = \mathbf{A}_{\mathsf{ID}}\mathbf{s} \right\},$$

  where $\rho := ([\mathbf{A}]_1, \{[\mathbf{B}_i]_1\}_{i\in[0,m]})$.

Remark 7 (Efficiency of the resulting scheme): While the above changes will increase the size of the master public and master private keys and worsen the reduction cost, they will not change the size of ciphertext and private keys, underlying assumptions, and leakage rate.

References

[1] P.C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," CRYPTO'99, ed. M.J. Wiener, LNCS, vol.1666, pp.388–397, Springer, Heidelberg, Aug. 1999.

[2] J.A. Halderman, S.D. Schoen, N. Heninger, W. Clarkson, W. Paul, J.A. Calandrino, A.J. Feldman, J. Appelbaum, and E.W. Felten, "Lest we remember: Cold boot attacks on encryption keys," USENIX Security 2008, ed. P.C. van Oorschot, pp.45–60, USENIX Association, July / Aug. 2008.

[3] A. Akavia, S. Goldwasser, and V. Vaikuntanathan, "Simultaneous hardcore bits and cryptography against memory attacks," TCC 2009, ed. O. Reingold, LNCS, vol.5444, pp.474–495, Springer, Heidelberg, March 2009.

[4] M. Naor and G. Segev, "Public-key cryptosystems resilient to key leakage," CRYPTO 2009, ed. S. Halevi, LNCS, vol.5677, pp.18–35, Springer, Heidelberg, Aug. 2009.

[5] Z. Brakerski, Y.T. Kalai, J. Katz, and V. Vaikuntanathan, "Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage," 51st FOCS, pp.501–510, IEEE Computer Society Press, Oct. 2010.

[6] Y. Dodis, K. Haralambiev, A. López-Alt, and D. Wichs, "Cryptography against continuous memory attacks," 51st FOCS, pp.511–520, IEEE Computer Society Press, Oct. 2010.

[7] A.B. Lewko, Y. Rouselakis, and B. Waters, "Achieving leakage resilience through dual system encryption," TCC 2011, ed. Y. Ishai, LNCS, vol.6597, pp.70–88, Springer, Heidelberg, March 2011.

[8] S.S.M. Chow, Y. Dodis, Y. Rouselakis, and B. Waters, "Practical leakage-resilient identity-based encryption from simple assumptions," ACM CCS 2010, ed. E. Al-Shaer, A.D. Keromytis, and V. Shmatikov, pp.152–161, ACM Press, Oct. 2010.

[9] K. Kurosawa and L.T. Phong, "Leakage resilient IBE and IPE under the DLIN assumption," ACNS 13, ed. M.J. Jacobson Jr., M.E. Locasto, P. Mohassel, and R. Safavi-Naini, LNCS, vol.7954, pp.487–501, Springer, Heidelberg, June 2013.

[10] Z. Yu, M.H. Au, Q. Xu, R. Yang, and J. Han, "Leakage-resilient functional encryption via pair encodings," ACISP 16, ed. J.K. Liu and R. Steinfeld, LNCS, vol.9722, pp.443–460, Springer, Heidelberg, July 2016.

[11] J. Zhang, J. Chen, J. Gong, A. Ge, and C. Ma, "Leakage-resilient attribute based encryption in prime-order groups via predicate encodings," Designs, Codes and Cryptography, vol.86, no.6, pp.1339–1366, 2018.

[12] M. Naor and M. Yung, "Public-key cryptosystems provably secure against chosen ciphertext attacks," 22nd ACM STOC, pp.427–437, ACM Press, May 1990.

[13] J. Alwen, Y. Dodis, M. Naor, G. Segev, S. Walfish, and D. Wichs, "Public-key encryption in the bounded-retrieval model," EUROCRYPT 2010, ed. H. Gilbert, LNCS, vol.6110, pp.113–134, Springer, Heidelberg, May / June 2010.

[14] Y. Dodis, K. Haralambiev, A. López-Alt, and D. Wichs, "Efficient public-key cryptography in the presence of key leakage," ASIACRYPT 2010, ed. M. Abe, LNCS, vol.6477, pp.613–631, Springer, Heidelberg, Dec. 2010.

[15] D. Hofheinz, D. Jia, and J. Pan, "Identity-based encryption tightly secure under chosen-ciphertext attacks," ASIACRYPT 2018, ed. T. Peyrin and S. Galbraith, LNCS, vol.11273, pp.190–220, Springer, Heidelberg, Dec. 2018.

[16] C.S. Jutla and A. Roy, "Shorter quasi-adaptive NIZK proofs for linear subspaces," ASIACRYPT 2013, ed. K. Sako and P. Sarkar, LNCS, vol.8269, pp.1–20, Springer, Heidelberg, Dec. 2013.

[17] S. Sun, D. Gu, and S. Liu, "Efficient leakage-resilient identity-based encryption with CCA security," PAIRING 2013, ed. Z. Cao and F. Zhang, LNCS, vol.8365, pp.149–167, Springer, Heidelberg, Nov. 2014.

[18] Y. Chen, B. Qin, and H. Xue, "Regularly lossy functions and applications," CT-RSA 2018, ed. N.P. Smart, LNCS, vol.10808, pp.491–511, Springer, Heidelberg, April 2018.

[19] T. Tomita, W. Ogata, and K. Kurosawa, "CCA-secure leakage-resilient identity-based key-encapsulation from simple (not q-type) assumptions," IWSEC 19, ed. N. Attrapadung and T. Yagi, LNCS, vol.11689, pp.3–22, Springer, Heidelberg, Aug. 2019.

[20] T. TOMITA, W. OGATA, K. KUROSAWA, and R. KUWAYAMA, "Cca-secure leakage-resilient identity-based encryption without q-type assumptions," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.E103.A, no.10, pp.1157–1166, 2020.

[21] Y. Zhou, B. Yang, Z. Xia, M. Zhang, and Y. Mu, "Identity-based encryption with leakage-amplified chosen-ciphertext attacks security," Theoretical Computer Science, vol.809, pp.277–295, 2020.

[22] A. Escala, G. Herold, E. Kiltz, C. Ràfols, and J. Villar, "An algebraic framework for Diffie-Hellman assumptions," CRYPTO 2013, ed. R. Canetti and J.A. Garay, LNCS, vol.8043, pp.129–147, Springer, Heidelberg, Aug. 2013.

[23] E. Kiltz and H. Wee, "Quasi-adaptive NIZK for linear subspaces revisited," EUROCRYPT 2015, ed. E. Oswald and M. Fischlin, LNCS, vol.9057, pp.101–128, Springer, Heidelberg, April 2015.

[24] C.S. Jutla and A. Roy, "Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces," CRYPTO 2014, ed. J.A. Garay and R. Gennaro, LNCS, vol.8617, pp.295–312, Springer, Heidelberg, Aug. 2014.

[25] B. Libert, T. Peters, M. Joye, and M. Yung, "Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures," EUROCRYPT 2014, ed. P.Q. Nguyen and E. Oswald, LNCS, vol.8441, pp.514–532, Springer, Heidelberg, May 2014.

[26] B. Libert, T. Peters, M. Joye, and M. Yung, "Compactly hiding linear spans - tightly secure constant-size simulation-sound QA-NIZK proofs and applications," ASIACRYPT 2015, ed. T. Iwata and J.H. Cheon, LNCS, vol.9452, pp.681–707, Springer, Heidelberg, Nov. / Dec. 2015.

[27] R. Langrehr and J. Pan, "Hierarchical identity-based encryption with tight multi-challenge security," PKC 2020, LNCS, pp.153–183, Springer, Heidelberg, 2020.

[28] R. Langrehr and J. Pan, "Unbounded HIBE with tight security," ASIACRYPT 2020, LNCS, pp.129–159, Springer, Heidelberg, Dec. 2020.

[29] O. Blazy, E. Kiltz, and J. Pan, "(Hierarchical) identity-based encryption from affine message authentication," CRYPTO 2014, ed. J.A. Garay and R. Gennaro, LNCS, vol.8616, pp.408–425, Springer, Heidelberg, Aug. 2014.

[30] J. Chen and J. Gong, "ABE with tag made easy - concise framework and new instantiations in prime-order groups," ASIACRYPT 2017, ed. T. Takagi and T. Peyrin, LNCS, vol.10625, pp.35–65, Springer, Heidelberg, Dec. 2017.

[31] P. Morillo, C. Ràfols, and J.L. Villar, "The kernel matrix Diffie-

Hellman assumption," ASIACRYPT 2016, ed. J.H. Cheon and T. Takagi, LNCS, vol.10031, pp.729–758, Springer, Heidelberg, Dec. 2016.
[32] M. Abe, M. Chase, B. David, M. Kohlweiss, R. Nishimaki, and M. Ohkubo, "Constant-size structure-preserving signatures: Generic constructions and simple assumptions," ASIACRYPT 2012, ed. X. Wang and K. Sako, LNCS, vol.7658, pp.4–24, Springer, Heidelberg, Dec. 2012.
[33] M. Abe, C.S. Jutla, M. Ohkubo, J. Pan, A. Roy, and Y. Wang, "Shorter QA-NIZK and SPS with tighter security," ASIACRYPT 2019, ed. S.D. Galbraith and S. Moriai, LNCS, vol.11923, pp.669–699, Springer, Heidelberg, Dec. 2019.
[34] R. Gay, D. Hofheinz, E. Kiltz, and H. Wee, "Tightly CCA-secure encryption without pairings," EUROCRYPT 2016, ed. M. Fischlin and J.S. Coron, LNCS, vol.9665, pp.1–27, Springer, Heidelberg, May 2016.
[35] M. Abe, C.S. Jutla, M. Ohkubo, and A. Roy, "Improved (almost) tightly-secure simulation-sound QA-NIZK with applications," ASIACRYPT 2018, ed. T. Peyrin and S. Galbraith, LNCS, vol.11272, pp.627–656, Springer, Heidelberg, Dec. 2018.
[36] B. Qin, S. Liu, and K. Chen, "Efficient chosen-ciphertext secure public-key encryption scheme with high leakage-resilience," IET Information Security, vol.9, no.1, pp.32–42, 2014.
[37] B. Waters, "Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions," CRYPTO 2009, ed. S. Halevi, LNCS, vol.5677, pp.619–636, Springer, Heidelberg, Aug. 2009.
[38] R. Cramer and V. Shoup, "Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption," EUROCRYPT 2002, ed. L.R. Knudsen, LNCS, vol.2332, pp.45–64, Springer, Heidelberg, April / May 2002.
[39] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," EUROCRYPT 2008, ed. N.P. Smart, LNCS, vol.4965, pp.146–162, Springer, Heidelberg, April 2008.
[40] Y. Ishai and H. Wee, "Partial garbling schemes and their applications," ICALP 2014, Part I, ed. J. Esparza, P. Fraigniaud, T. Husfeldt, and E. Koutsoupias, LNCS, vol.8572, pp.650–662, Springer, Heidelberg, July 2014.
[41] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," EUROCRYPT 2004, ed. C. Cachin and J. Camenisch, LNCS, vol.3027, pp.523–540, Springer, Heidelberg, May 2004.

Appendix A: Construction: USS QA-NIZK Argument for GTLS

Here, we show a *USS* QA-NIZK argument Π_{USS} for $\mathcal{L}^{\mathsf{GTLS}}$. Our QA-NIZK $\Pi_{\mathsf{USS}} = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ is defined in Figure A· 1. Our construction is based on the USS QA-NIZK argument proposed by Kiltz and Wee [23].

Theorem 4: Π_{USS} defined in Figure A· 1 has perfect completeness and perfect zero-knowledge. Furthermore, if the the $\mathcal{D}_k$-MDDH problem in $\mathbb{G}_1$ and the $\mathcal{D}_k$-KerMDH problem in $\mathbb{G}_2$ is hard and $\mathcal{H}$ is a CR hash function family, then Π_{USS} has unbounded simulation-soundness.

We only give the proof overview of this theorem because our proof is similar to [23, Theorem 4].

Proof Overview. Perfect completeness and perfect zero-knowledge follow readily from the fact that for all $\mathbf{c} = \mathbf{M}_{\mathbf{x}}\mathbf{r}$, we have $\mathbf{P}_{\mathbf{x}}^\top \mathbf{r} = \sum_{i=1}^m x_i \mathbf{P}_i^\top \mathbf{r} = \mathbf{K}_{\mathbf{x}}^\top \mathbf{M}_{\mathbf{x}}\mathbf{r} = \mathbf{K}_{\mathbf{x}}^\top \mathbf{c}$ and for all $\mathbf{A}$, $\mathbf{B}$, $\mathbf{w}$, $\mathbf{K}_0'$ and $\mathbf{K}_1'$, we have

$$[\mathbf{w}^\top \mathbf{B}^\top (\mathbf{K}_0' + \tau \mathbf{K}_1')]_1 \circ [\mathbf{A}]_2 = [\mathbf{w}^\top \mathbf{B}^\top]_1 \circ [\mathbf{K}_0'\mathbf{A} + \tau \mathbf{K}_1'\mathbf{A}]_2,$$

$\underline{\mathsf{Gen}(\mathsf{par}, \rho = ([\mathbf{M}]_1, [\mathbf{M}_1']_1, \ldots, [\mathbf{M}_m']_1)):}$
$\mathsf{H} \leftarrow\!\$\, \mathcal{H}$, $\mathbf{A}, \mathbf{B} \leftarrow\!\$\, \mathcal{D}_k \subset \mathbb{Z}_q^{(k+1)\times k}$
$\mathbf{K} \leftarrow\!\$\, \mathbb{Z}_q^{n'\times(k+1)}$, $\mathbf{K}_0', \mathbf{K}_1' \leftarrow\!\$\, \mathbb{Z}_q^{(k+1)\times(k+1)}$
$\mathbf{Y} := \mathbf{K}\mathbf{A}$, $(\mathbf{Y}_0', \mathbf{Y}_1') := (\mathbf{K}_0'\mathbf{A}, \mathbf{K}_1'\mathbf{A})$, $(\mathbf{P}_0', \mathbf{P}_1') := (\mathbf{B}^\top \mathbf{K}_0', \mathbf{B}^\top \mathbf{K}_1')$
For $i = 1, \ldots, m$:
  $\mathbf{K}_i \leftarrow\!\$\, \mathbb{Z}_q^{n\times(k+1)}$, $\mathbf{Y}_i := \mathbf{K}_i\mathbf{A} \in \mathbb{Z}_q^{n\times k}$
  $[\mathbf{P}_i]_1 := [\mathbf{M}^\top \mathbf{K}_i + \mathbf{M}_i'^\top \mathbf{K}]_1$
$\mathsf{crs} := (\{[\mathbf{P}_i]_1, [\mathbf{Y}_i]_2\}_i, [\mathbf{P}_0']_1, [\mathbf{P}_1']_1, [\mathbf{Y}]_2, [\mathbf{Y}_0']_2, [\mathbf{Y}_1']_2, [\mathbf{A}]_2, [\mathbf{B}]_1, \mathsf{H})$
$\mathsf{td} := (\mathbf{K}, \{\mathbf{K}_i\}_i)$
Return $(\mathsf{crs}, \mathsf{td})$

$\underline{\mathsf{Prove}(\mathsf{crs}, ([\mathbf{c}]_1, \mathbf{x}), \mathbf{r}):}$ // $\mathbf{c} = \mathbf{M}_{\mathbf{x}}\mathbf{r}$
$\mathbf{w} \leftarrow\!\$\, \mathbb{Z}_q^k$, $[\mathbf{t}]_1 := [\mathbf{B}\mathbf{w}]_1$, $\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x}, [\mathbf{t}]_1) \in \mathbb{Z}_q$
$[\mathbf{P}_{\mathbf{x}}]_1 := \left[\sum_{i=1}^m x_i \mathbf{P}_i\right]_1$, $[\mathbf{u}]_1 := [\mathbf{P}_{\mathbf{x}}^\top \mathbf{r} + (\mathbf{P}_0' + \tau \mathbf{P}_1')^\top \mathbf{w}]_1$
Return $\pi := ([\mathbf{u}]_1, [\mathbf{t}]_1) \in (\mathbb{G}_1^{k+1})^2$

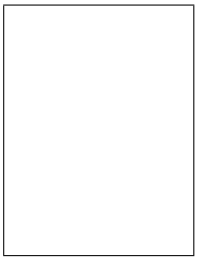
$\underline{\mathsf{Ver}(\mathsf{crs}, ([\mathbf{c}]_1, \mathbf{x}), \pi =: ([\mathbf{u}]_1, [\mathbf{t}]_1)):}$
$\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x}, [\mathbf{t}]_1) \in \mathbb{Z}_q$, $[\mathbf{Y}_{\mathbf{x}}]_2 := \begin{bmatrix} \sum_{i=1}^m x_i \mathbf{Y}_i \\ \mathbf{Y} \end{bmatrix}_2$
If $[\mathbf{u}]_1^\top \circ [\mathbf{A}]_2 = [\mathbf{c}]_1^\top \circ [\mathbf{Y}_{\mathbf{x}}]_2 + [\mathbf{t}]_1^\top \circ [\mathbf{Y}_0' + \tau \mathbf{Y}_1']_2$: Return 1
Else: Return 0

$\underline{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, ([\mathbf{c}]_1, \mathbf{x})):}$
$\mathbf{w} \leftarrow\!\$\, \mathbb{Z}_q^k$, $[\mathbf{t}]_1 := [\mathbf{B}\mathbf{w}]_1$, $\tau := \mathsf{H}([\mathbf{c}]_1, \mathbf{x}, [\mathbf{t}]_1) \in \mathbb{Z}_q$
$\mathbf{K}_{\mathbf{x}} := \begin{pmatrix} \sum_{i=1}^m x_i \mathbf{K}_i \\ \mathbf{K} \end{pmatrix}$, $[\mathbf{u}]_1 := [\mathbf{K}_{\mathbf{x}}^\top \mathbf{c} + (\mathbf{P}_0' + \tau \mathbf{P}_1')^\top \mathbf{w}]_1$
Return $\pi := ([\mathbf{u}]_1, [\mathbf{t}]_1) \in (\mathbb{G}_1^{k+1})^2$

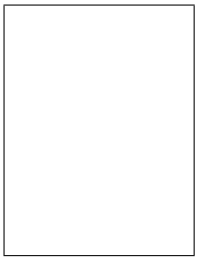
Fig. A· 1 Our USS QA-NIZK argument Π_{USS}.

where $\tau = \mathsf{H}([\mathbf{c}]_1, \mathbf{x}, [\mathbf{t}]_1)$.

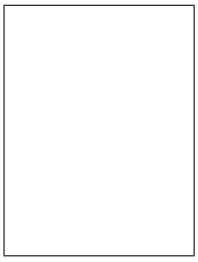
Next, we consider USS. Similar to the USS QA-NIZK in [23], we can show that no information on $\mathbf{K}$ and $\{\mathbf{K}_i\}_i$ is leaked from all simulated proofs thanks to the technique of using pseudorandom MAC [29]. If information on $\mathbf{K}$ and $\{\mathbf{K}_i\}_i$ does leak only from crs, then $\mathbf{K}_{\mathbf{x}^*}\mathbf{c}^*$ for $\mathbf{c}^* \notin \mathsf{Span}(\mathbf{M}_{\mathbf{x}^*})$ is uniformly random from the view point of an adversary, that guarantees the unbounded simulation soundness. □



Toi Tomita received the B. E. and M. E. degrees in information and communication engineering in 2017 and 2019, respectively, from Tokyo Institute of Technology. He is currently a doctor course student in Tokyo Institute of Technology. He is also a research assistant in the National Institute of Advanced Industrial Science and Technology (AIST). His current interests are cryptography and information security.



Wakaha Ogata received the B. S., M. E. and D. E. degrees in electrical and electronic engineering in 1989, 1991 and 1994, respectively, from Tokyo Institute of Technology. From 1995 to 2000, she was an Assistant Professor at Himeji Institute of Technology. Since 2000 she has been working for Tokyo Institute of Technology, and now she is a Professor from 2013. Her current interests are cryptography and information security.



Kaoru Kurosawa received the B.E. and Dr. Eng. degrees in electrical engineering in 1976 and 1981, respectively, from Tokyo Institute of Technology. He is a Professor Emeritus at Ibaraki University. He is also a visiting researcher of National Institute of Advanced Industrial Science and Technology and Research and Development Initiative, Chuo University. His current research interest is cryptography. He was Program Chair for Asiacrypt 2007, PKC 2013 and some other conferences. Dr. Kurosawa is a Fellow of IACR and IEICE, and a member of IEEE. He received the excellent paper award of IEICE in 1981, the young engineer award of IEICE in 1986, Telecom System Scientific Award of Telecommunications Advancement Foundation in 2006 and Achievement Award of IEICE in 2007.