PAPER

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

Toi TOMITA†, 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
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\(^\dagger\). 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.

We summarize the differences between our QA-NIZK argument and the existing ones in Table 1.

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

\[
\mathcal{L}_\rho^{\text{GTLS}} := \{([c], x) \mid \exists r \in \mathbb{Z}_q^t \text{ s.t. } c = M_x r\},
\]

where \(\rho := ([\mathcal{M}], [\mathcal{M}^1], \ldots, [\mathcal{M}^m]) \in \mathbb{G}^{n \times t} \times (\mathbb{G}^{n \times t})^m\), \(n > t\), \(n' \geq 1\), \(x_i\) is the \(i\)-th element of \(x \in \mathbb{Z}_q^m\), and \(M_x := \left(\sum_{i=1}^m M^M_{i,x} \right)\). (We use implicit representation of group elements as in [22].) 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:

\[
\mathcal{L}_\rho^{\text{GTLS}} := \{([c], L) \mid \exists r \in \mathbb{Z}_q^t \text{ s.t. } c = M_L r\},
\]

where \(L\) is a linear map and \(M_L := \left(M_{L,\ldots,M_{L,m}}\right)\). 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.

\(^\dagger\)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-CCA2-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 viewpoint. 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:

\[
\mathcal{L}_\rho^{\text{tagged}} := \{([c], x) \mid \exists r \in \mathbb{Z}_q^t \text{ s.t. } c = M_x r\},
\]

where \(\rho := ([\mathcal{M}], [\mathcal{M}^1], [\mathcal{M}^2])\) and \(M_x := \left(M_{x,\ldots,M_{x,m}}\right)\). In the DV variant of Jutla-Roy’s scheme, a verifier has a secret verification key which is random vectors \(k \in \mathbb{Z}_q^t\) and \(k_0, k_1 \in \mathbb{Z}_q^{\rho_1}\), and the common reference string (CRS) is the projections
spaces (GTLS) given by Eq. (1), where QA-NIZK argument for the generalized statement.

a by using their techniques, the scheme can be converted to LR-CCA2-secure ABE schemes.

language in Eq. (2). We refer to Section 3.3 for details.

that a QA-NIZK argument for GTLS implies one for the argument can be extended to the GTLS in this paper. This is one of our contribution. We further show that this

\[
\begin{align*}
\mathbf{B} &= \mathbf{M}^\top \mathbf{k} \\
\mathbf{M} &= \begin{bmatrix} 0 & 1 \end{bmatrix}
\end{align*}
\]

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 3.5 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 \([\mathbf{c}]\).

### 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.

<table>
<thead>
<tr>
<th>Linear subspaces ((m = 0))</th>
<th>Tagged linear subspaces ((m \geq 1))</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-simulation-sound</td>
<td>Already exist, e.g., JR13 [16]</td>
</tr>
<tr>
<td>Simulation-sound</td>
<td>Already exist, e.g., KW15 [23]</td>
</tr>
</tbody>
</table>

#### 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 \( \varepsilon \) and the empty set \( \emptyset \).
- We use \( x \leftarrow S \) to denote the process of sampling an element \( x \) from \( S \) uniformly at random if \( S \) is a finite set. We denote the bit length of element \( x \) as \( \|x\| \).
- We denote a security parameter as \( \lambda \).

#### Games

Following [29], we use code-based games to define and prove security. A game contains procedures \( \mathcal{A}, \mathcal{B} \) and the empty set \( \emptyset \). We denote the empty string as \( \emptyset \).

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 \( b \).

#### Collision Resistant Hash Functions

Let \( \mathcal{H} \) be a family of hash functions \( \mathcal{H} : \{0, 1\}^* \to X \), where \( X = \mathbb{X}_1 \) 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 \( \mathcal{A} \),

\[
\text{Adv}^\text{CR}_{\mathcal{H}, \mathcal{A}} (\lambda) := \Pr[ x \neq x' \land H(x) = H(x') \mid H \leftarrow \mathcal{H}, (x, x') \leftarrow \mathcal{A}(1^\lambda, H)]
\]
is negligible.

2.1 Pairing Groups and Matrix Diffie-Hellman Assumptions

Let \( \text{GGen} \) be a PPT algorithm that on input \( 1^t \) returns a description \( G = (G_1, G_2, G_T, p, P_1, P_2, e) \) of asymmetric pairing groups, where \( G_1, G_2, G_T \) are cyclic-groups of order \( q \) for a \( t \)-bit prime \( q \), and \( e : G_1 \times G_2 \rightarrow G_T \) is an efficient computable (non-degenerated) bilinear map. Define \( P_T \) as \( e(P_1, P_2) \), which is a generator in \( G_T \).

We use implicit representation of group elements as in \([22]\). For \( s \in \mathbb{Z} \), we have \( e(P_1, P_2) \) efficiently computed given \( a, b \in \mathbb{Z} \). Because they essentially share \( n \times ℓ \) and \( \ell \) rows of \( \mathbb{Z} \), we denote \( \mathbb{Z} \) as the implicit representation of \( G \).

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

Definition 2 (Matrix distribution): Let \( k, ℓ \in \mathbb{N} \) with \( ℓ > k \). We call \( D_{ℓ,k} \) a matrix distribution if it outputs matrices in \( \mathbb{Z}^{ℓ \times k} \) of full rank \( k \) in polynomial time. By \( D_k \), we denote \( D_{k+1,k} \).

Without loss of generality, we assume the first \( k \) rows of \( A \leftarrow D_{ℓ,k} \) (i.e., \( A \in \mathbb{Z}^{ℓ \times k} \)) form an invertible matrix. For a matrix \( A \leftarrow D_{ℓ,k} \), we define the set of kernel of \( A \) as
\[
\ker(A) := \{ A^+ \in \mathbb{Z}^{ℓ \times (ℓ−k)} \mid A^+A^+ = 0 \in \mathbb{Z}^{ℓ \times (ℓ−k)} \}
\]
and \( A^+ \) has rank \( (ℓ−k) \).

Given a matrix \( A \) over \( \mathbb{Z}^{ℓ \times k} \), it is efficient to sample an \( A^+ \) from \( \ker(A) \).

The \( D_{ℓ,k} \)-Matrix Diffie-Hellman problem is to distinguish the two distributions \( \{ [A]_s, [Aw]_s \} \) and \( \{ [A]_s, [u]_s \} \), where \( A \leftarrow D_{ℓ,k} \), \( w \leftarrow \mathbb{Z}^{ℓ \times k} \), and \( u \leftarrow \mathbb{Z}^{ℓ \times k} \).

Definition 3 (\( D_{ℓ,k} \)-Matrix Diffie-Hellman assumption): Let \( k \geq 1 \) and \( ℓ > k \) be integers. Let \( D_{ℓ,k} \) be a matrix distribution and \( s \in \{1,2,T\} \). We say that the \( D_{ℓ,k} \)-Matrix Diffie-Hellman (\( D_{ℓ,k} \)-MDDH) problem is hard relative to \( \text{GGen} \) in group \( G_s \) if for any PPT adversary \( \mathcal{A} \),
\[
\text{Adv}^{\text{MDDH}}_{G_s, D_{ℓ,k}, \mathcal{A}}(\lambda) := \text{Pr}[1 \leftarrow \mathcal{A}(G, [A]_s, [Aw]_s)] - \text{Pr}[1 \leftarrow \mathcal{A}(G, [A]_s, [u]_s)]
\]
is negligible, where the probability is taken over \( \mathcal{G} \leftarrow \text{GGen}(1^\lambda) \), \( A \leftarrow D_{ℓ,k} \), \( w \leftarrow \mathbb{Z}^{ℓ \times k} \), and \( u \leftarrow \mathbb{Z}^{ℓ \times k} \).

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

Definition 4 (\( D_{ℓ,k} \)-Kernel Diffie-Hellman assumption): Let \( k \geq 1 \) and \( ℓ > k \) be integers. Let \( D_{ℓ,k} \) be a matrix distribution and \( s \in \{1,2\} \). We say that the \( D_{ℓ,k} \)-Kernel Diffie-Hellman (\( D_{ℓ,k} \)-KerMDH) problem is hard relative to \( \text{GGen} \) in group \( G_s \) if for any PPT adversary \( \mathcal{A} \),
\[
\text{Adv}^{\text{KerMDH}}_{G_s, D_{ℓ,k}, \mathcal{A}}(\lambda) := \text{Pr}[c^T A = 0 \land c \neq 0 \mid [c]_{3-s} \leftarrow \mathcal{A}(G, [A]_s)]
\]
is negligible, where the probability is taken over \( \mathcal{G} \leftarrow \text{GGen}(1^\lambda) \), \( A \leftarrow D_{ℓ,k} \).

The following lemma shows that the \( D_{ℓ,k} \)-KerMDH assumption is a relaxation of the \( D_{ℓ,k} \)-MDDH assumption.

Lemma 1 (\( D_{ℓ,k} \)-MDDH \( \Rightarrow \) \( D_{ℓ,k} \)-KerMDH \([31]\) ): For any matrix distribution \( D_{ℓ,k} \), if \( D_{ℓ,k} \)-MDDH in group \( G_s \) is hard, then \( D_{ℓ,k} \)-KerMDH in group \( G_s \) is hard.

We also define the external \( D_{ℓ,k} \)-matrix Diffie-Hellman (\( D_{ℓ,k} \)-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 \( D_{ℓ,k} \)-MDDH assumption): Let \( k \geq 2 \) and \( ℓ > k \) be integers. Let \( D_{ℓ,k} \) be a matrix distribution
and $s \in \{1, 2\}$. We say that the external $D_{\ell,k}$-Matrix Diffie-Hellman $(D_{\ell,k}$-exMDH) problem is hard relative to $GG\text{Gen}$ in group $G_3$, if for any PPT adversary $A$, 
$$\text{Adv}^{\text{exMDH}}_{G_3, D_{\ell,k}, A}(\lambda) \triangleq \Pr[1 \leftarrow s A(G, [A]_1, [A]_2, [Aw]_3) \leftarrow 1 - \Pr[1 \leftarrow s A(G, [A]_1, [A]_2, [u]_3)]$$
is negligible, where the probability is taken over $G \leftarrow GG\text{Gen}(\lambda^4)$, $A \leftarrow D_{\ell,k}$, $w \leftarrow \mathbb{Z}^k_q$, and $u \leftarrow \mathbb{Z}_q^\ell$.

### 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 generalized NIZK argument used to boost CPA to CCA2 for leakage-on the language for which proofs are generated.

#### 3.1 Definition

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

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

- $\text{Gen}(\text{par}, \rho) \rightarrow (\text{crs}, \text{td})$: The generation algorithm takes as input the public parameter and a string $\rho \in D_{\text{par}}$. It outputs a common reference string $\text{crs}$ and a trapdoor $\text{td}$.
- $\text{Prove}(\text{crs}, y, w) \rightarrow \pi$: The proving algorithm takes as input the $\text{crs}$, a statement $y$, and a witness $w$ with $R_{\rho}(y, w) = 1$, and outputs a proof $\pi$.

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

Here, we show our OT-SS QA-NIZK argument for $L_{\text{GTLS}} := \{L_{\rho}^{\text{GTLS}}\}_{\rho \in D_{\text{par}}}$.

**Fig. 1** USS security game for QA-NIZK

1, and outputs a proof $\pi$.

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

- $\text{Sim}(\text{crs}, \text{td}, y) \rightarrow \pi$: The simulation algorithm takes as input $\text{crs}$, $\text{td}$, and a statement $y$ (not necessarily in $L_{\rho}$), and outputs a simulated proof $\pi$.

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

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

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

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

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

**Remark 1** (Variants of definitions of simulation-soundness): Here, we use a stronger version of simulation-soundness used in [15], [33] that requires $(y^*, \pi^*) \notin Q_{\text{Sim}}$, rather than the weaker version used in [23], [34], [35] that only requires $y^* \notin Q_{\text{Sim}}$. 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.

#### 3.2.1 Construction: OT-SS QA-NIZK Argument for GTLS

Here, we show our OT-SS QA-NIZK argument for $L_{\text{GTLS}} := \{L_{\rho}^{\text{GTLS}}\}_{\rho \in D_{\text{par}}}$. Let $H = \{H : \{0, 1\} \rightarrow \mathbb{Z}_q\}$ be a CR hash function family. Our QA-NIZK argument $\Pi_{\text{OT-SS}} = (\text{Gen}, \text{Prove}, \text{Ver}, \text{Sim})$ is defined in Figure 2.
our construction can be easily extended to a tag-based QA-NIZK argument by adding the label lbd to the input of hash function. Thus, our construction can be used in all the applications that require tag-based QA-NIZK arguments.

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

Proof. To prove the lemma, fix matrices $M \in \mathbb{Z}_q^{n \times t}$, $M'_1, \ldots, M'_m \in \mathbb{Z}_q^{n' \times t}$, and $A \in \mathbb{Z}_q^{(k+1) \times k}$, and pick a non-zero vector $\hat{a} \notin \text{Span}(A)$. For any $x' \in \mathbb{Z}_q^n$ and $c' \in \mathbb{Z}_q^{n+n'}$ such that $c' \notin \text{Span}(M_x')$, there exist vectors $m_1^1, \ldots, m_m^1 \in \mathbb{Z}_q^n$ and $m^+ \in \mathbb{Z}_q^n$ such that
\[
\begin{pmatrix}
M^T \\
O & M^T \\
\vdots & \ddots & \ddots & \ddots \\
O & \cdots & O & M^T & m_1^1 \\
O & \cdots & O & \cdots & m_m^1 \\
(x_1^c)^T & x_2^c)^T & \cdots & x_m^c)^T
\end{pmatrix}
\begin{pmatrix}
m_1^T \\
m_2^T \\
\vdots \\
m_m^T \\
m^+ \\
\theta
\end{pmatrix}
= 0
\]
\[
\begin{pmatrix}
\theta \\
\vdots \\
\theta
\end{pmatrix}
\]
(4)

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

Proof. Perfect completeness and perfect zero-knowledge follow readily from the fact that
\[
\begin{pmatrix}
P_x^{(0)} + \tau P_x^{(1)}
\end{pmatrix}^T
= \sum_{i=1}^m (x_i P_i^{(0)} + \tau x_i P_i^{(1)})^T
\]
\[
= \sum_{i=1}^m (x_i (K_i^{(0)})^T M + K_i^{(0)} M_i^T)^T + \tau x_i (K_i^{(1)})^T M_i^T M_i^T
\]
\[
= \sum_{i=1}^m x_i K_i^{(0)} + \tau \sum_{i=1}^m x_i M_i^T M_i
\]
\[
= K_h^{(0)} + \tau K_h^{(1)} M_x
\]
\[
\begin{align*}
\text{INIT}(\mu): & \quad \text{for } G_0 \rightarrow G_2 \\
H \rightarrow S \mathcal{H} & \quad A \rightarrow \mathcal{D}_k \subset \mathbb{Z}_q^{(k+1) \times k} \\
0 \rightarrow 2n_q^{(k+1)} & \quad K^{(0)} \rightarrow \mathbb{Z}_q^{n_q^{(k+1)}} \\
\mathbf{Y}(0), \mathbf{Y}(1) & \rightarrow (K^{(0)}, A, K^{(1)}(A)) \\
\text{for } i = 1, \ldots, m: & \quad K_i^{(0)}, K_i^{(1)} \rightarrow \mathbb{Z}_q^{n_q^{(k+1)}} \\
\mathbf{Y}(0), \mathbf{Y}(1) & \rightarrow (K_i^{(0)}, A, K_i^{(1)}(A)) \\
\mathbf{P}^{(b)}_i & \rightarrow [M_i \mathbf{K}_i^{(b)} + M_i^* \mathbf{K}_i^{(b)}]_i \quad \text{for } b \in \{0, 1\} \\
\text{crs} & \rightarrow (\mathbb{P}^{(b)}_i, \mathbb{Y}(b)_2, \mathbb{A}_2, H) \\
\text{Return crs} \\
\end{align*}
\]

\[
\begin{align*}
\text{SIM}(\{e\}, x_1): & \quad \text{for } G_0 \rightarrow G_2 \\
\tau & \leftarrow \text{H}(\{e\}, x_1), \mathbf{K}_i^{(b)} \leftarrow \left(\sum_{i=1}^{m} \mathbf{K}_i^{(b)}\right) \quad \text{for } b \in \{0, 1\} \\
\pi & \leftarrow (\mathbf{K}_i^{(0)} + \tau \mathbf{K}_i^{(1)})^T \mathbf{c}_i \in \mathbb{G}_k^{n_k^1} \\
\text{Return } \pi \\
\end{align*}
\]

\[
\begin{align*}
\text{FINALIZE}((\{e\}, x_1), \pi^* = [u_1^*]): & \quad \text{for } G_0 \rightarrow G_2 \\
\mathbf{H}(\{e\}, x_1), \mathbf{Y}(b)_2 & \leftarrow \left(\sum_{i=1}^{m} \mathbf{K}_i^{(b)}\right) \quad \text{for } b \in \{0, 1\} \\
\tau^* & \leftarrow \mathbf{H}(\{e\}, x_1), [\mathbf{Y}(b)_2]_b \leftarrow \left(\sum_{i=1}^{m} \mathbf{K}_i^{(b)}\right) \quad \text{for } b \in \{0, 1\} \\
\text{If } \tau^* = \tau: \text{Return } 0 \\
\text{If } \tau^* \neq \tau: \text{Return } 0 \\
\end{align*}
\]

Fig. 4 Games G0, G1, and G2 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.

\[
\begin{align*}
0 \rightarrow (K_x^{(0)} + \tau K_x^{(1)})^T M_x, \\
\text{and for all } c = M_x r, \\
\mathbf{P}(0) + \tau \mathbf{P}(0)^T & = (K_x^{(0)} + \tau K_x^{(1)})^T M_x r \\
\end{align*}
\]

where \( \tau \leftarrow \text{H}(\{e\}, x) \).

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

\[
\text{Adv}_{\Pi_{\text{OT-SS}}, \mathcal{A}}(\lambda) \leq \text{Adv}_{\text{KerMDH}}^{\text{G}_2, \mathcal{D}_k, \mathcal{B}}(\lambda) + \text{Adv}_{\mathcal{H}, \mathcal{B}}^{\text{CR}}(\lambda) + 1/q. \\
\]

We bound the advantage of \( \mathcal{A} \) via a sequence of games defined in Figure 4. \( \mathcal{G}_0 \) is the real OT-SS game for QA-NIZK as defined in Figure 1.

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

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

**Proof.** \( \mathcal{G}_1 \) is identical to \( \mathcal{G}_0 \) unless \( \mathcal{A} \) queries FINALIZE with \((\{e\}, x^*), [u_1^*] \) such that \([u_1^*] \neq (K_x^{(0)} + \tau K_x^{(1)})^T [e_1] \) is a non-zero vector in ker(A), which is a solution of the \( \mathcal{D}_k \)-KerMDH problem for \( \mathcal{A} \). Hence, we have \( |\text{Pr}[\mathcal{G}_0^{\mathcal{A}} \Rightarrow 1] - \text{Pr}[\mathcal{G}_1^{\mathcal{A}} \Rightarrow 1]| \leq \text{Adv}_{\text{KerMDH}}^{\text{G}_2, \mathcal{D}_k, \mathcal{B}}(\lambda) \).

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

**Proof.** The difference between \( \mathcal{G}_1 \) and \( \mathcal{G}_2 \) happens when \( \mathcal{A} \) queries FINALIZE with \((\{e\}, x^*), \pi^* \) such that \((\{e\}, x^*), \pi^* \) \neq \((\{e\}, x, \pi) \) and \( \tau^* = \tau \). To bound this, we consider the following cases:

- \((\{e\}, x^*), \pi^* \) \neq \((\{e\}, x, \pi) \) 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 \( \text{Adv}_{\mathcal{H}, \mathcal{B}}^{\text{CR}}(\lambda) \).

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

**Lemma 6 (\( \mathcal{G}_2 \)):** \( \text{Pr}[\mathcal{G}_2^{\mathcal{A}} \Rightarrow 1] \leq 1/q. \)

**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 \( \Pi_{\text{OT-SS}} \), then \( \mathcal{B} \) perfectly simulates \( \mathcal{G}_2 \). Thus, we have \( \text{Pr}[\mathcal{G}_2^{\mathcal{A}} \Rightarrow 1] = \text{Pr}[\text{Core}_{\text{OT-SS}}^{\mathcal{B}} \Rightarrow 1] \). From Lemma 2, we have \( \text{Pr}[\mathcal{G}_2^{\mathcal{A}} \Rightarrow 1] \leq 1/q \).

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 \( \rho := ([M_1, [M'_1]_1, \ldots, [M'_{m_1}]_1]) \in \mathbb{G}^{n \times t} \times (\mathbb{G}'^{n \times t})^{m_1} \). For any \( \mathbb{Z}_q \)-linear map \( L : (\mathbb{Z}_q^{n \times t})^m \rightarrow \mathbb{Z}_q^{n' \times t} \), we define

\[
\hat{L}_p := \left\{ ([c], 1) \mid \exists r \in \mathbb{Z}_q^t \text{ s.t. } c = M_L r \right\},
\]

where \( M_L := (L(M'_1, \ldots, M'_{m_1})) \). Note that a linear subspace to which \( e \) should belong depends on a linear map \( L \), but not on a vector \( x \). We can see that our QA-NIZK arguments support the above language \( \hat{L}_p \) as follows: If we express \( L \) as \( L = (i_{i,j}) \in \mathbb{Z}_q^{m \times m'} \), we have

\[
L(M'_1, \ldots, M'_{m_1}) = \left( \sum_{i=1}^{m} l_i, M'_1 \right) \cdots \left( \sum_{i=1}^{m} l_i, M'_{m_1} \right) = \sum_{i=1}^{m} l_i \cdot M'_i = \left( \sum_{i=1}^{m} l_i, \hat{M}'_{i, j} \right),
\]

where \( \hat{M}'_{i, j} \in \mathbb{Z}_q^{n'\times t} \) is the matrix whose coordinates are all zero. Therefore, we can appropriately determine \( \hat{M}'_{i, j, m_1} \in \mathbb{Z}_q^{n'\times t} \) from \( M'_1, \ldots, M'_{m_1} \) that satisfy \( L(M'_1, \ldots, M'_{m_1}) = \sum_{i=1}^{m_1} \sum_{j=1}^{m'_{i,j}} M_{i,j} \). Then, we have

\[
\hat{L}_p \subset L_p \subset L_p^{\text{GTLS}},
\]

where \( \rho := (\{M_1, [M'_1]_1, \ldots, [M'_{m_1}]_1\}) \in \mathbb{G}_1^{n \times t} \times (\mathbb{G}'_1^{n \times t})^{m_1} \) and \( l := (l_1, \ldots, l_{m_1}) \in \mathbb{Z}_q^{m \times m'} \).

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.
INIT:
\[ b \leftarrow \mathcal{S}(\{0,1\}); (\text{mpk}, \text{msk}) \leftarrow \text{Setup}(1^t, X, Y) \]
\[ Q_T := \{(0, \epsilon, \text{msk}, 0)\} \]
Return mpk

CREATE\(h, y\):
Find \((h, y', \text{sk}_y, L) \in Q_T\)
If \(y' = \epsilon\) (i.e., \(\text{sk}_y = \text{msk}\)):
\[ \text{sk}_y \leftarrow \mathcal{S}\text{Gen}(\text{mpk}, y) \]
\[ Q_T := Q_T \cup \{(H + 1, y, \text{sk}_y, 0)\}; H := H + 1 \]
Return \(\bot\)

REVEAL\((h)\):
Find \((h, y, \text{sk}_y, L) \in Q_T\)
If \(y \in Q_L\) s.t. \(P(x', y) = 1\): Return \(\bot\)
If \(y \neq \epsilon\):
\[ Q_T := Q_T \cup \{y\} \]
Return \(\text{sk}_y\)
Else: Return \(\bot\)

LEAK\((h, f)\):
If \(f(g) = 1\): Return \(\bot\)
Find \((h, y, \text{sk}_y, L) \in Q_T\)
If \(y \neq \epsilon\) and \(L + |f(\text{sk}_y)| \leq \ell_{sk}\):
\[ L := L + |f(\text{sk}_y)| \]
Return \(f(\text{sk}_y)\)
If \(y = \epsilon\) (i.e., \(\text{sk}_y = \text{msk}\)) and \(L + |f(\text{msk})| \leq \ell_{msk}\):
\[ L := L + |f(\text{msk})| \]
Return \(f(\text{msk})\)

UPDATE\((h)\):
Find \((h, y, \text{sk}_y, L) \in Q_T\)
If \(y = \epsilon\) (i.e., \(\text{sk}_y = \text{msk}\)):
\[ \text{msk} \leftarrow \mathcal{S}\text{UpdateMSK}(\text{mpk}, \text{msk}) \]
\[ Q_T := Q_T \cup \{(H + 1, \epsilon, \text{msk}, 0)\}; H := H + 1 \]
Else:
\[ \text{sk}_y \leftarrow \mathcal{S}\text{UpdateSK}(\text{mpk}, \text{sk}_y, y) \]
\[ Q_T := Q_T \cup \{(H + 1, y, \text{sk}_y, 0)\}; H := H + 1 \]
Return \(\bot\)

DECAP\((h, c)\):
Find \((h, y, \text{sk}_y, L) \in Q_T\)
If \(y = \epsilon\) (i.e., \(\text{sk}_y = \text{msk}\)): Return \(\bot\)
If \(c_t \neq ct' \text{ or } R(x', y) = 0\): Return Decap\((\text{sk}_y, y, ct)\)
Return \(\bot\)

CHAL\((x')\):
\[ \ell := 1 \]
If \(y \in Q_L\) s.t. \(P(x', y) = 1\): Return \(\bot\)
\[ (ct', K')_y \leftarrow \text{Encap}(\text{mpk}, x'); K' \leftarrow K \]
Return \((ct', K')_y\)

FINALIZE\((h)\):
Return \((h, \ell, L^A)\)

**Fig. 6** Security game LR-CCA\(_{\text{CML}}\).

ABKEM is \((\ell_{msk}, \ell_{sk})\)-LR-CCA\(_{\text{C2}}\)-secure if for any PPT adversary \(\mathcal{A}\), 
\[
\text{Adv}_{\text{ABKEM} \mathcal{A}, LR-CCA\(_{\text{C2}}\)}(L) := \left| \text{Pr}[\text{ABKEM} \mathcal{A}, LR-CCA\(_{\text{C2}}\) \Rightarrow 1] - 1/2 \right| \ 	ext{is negligible, where Game LR-CCA\(_{\text{CML}}\) 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 \(P : X \times Y \rightarrow \{0,1\}\) be a predicate.

An LRPE for \(P\) is a tuple of deterministic algorithms \((\text{sE}, \text{mE}, \text{mkE}, \text{rE}, \text{rKE}, \text{sD}, \text{dD})\) as follows: \(\text{sE} : X \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^n, \text{mE} : \mathbb{Z}_q^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^{nq}, \text{mkE} : \mathbb{Z}_q^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^{nq}, \text{rE} : Y \times \mathbb{Z}_q^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^{nq}, \text{rKE} : Y \times \mathbb{Z}_q^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^{nq}, \text{sD} : X \times Y \times \mathbb{Z}_q^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^n, \text{dD} : X \times Y \times \mathbb{Z}_q^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_q^n\) for some \(n, n_1, n_r, n_m, n_z \in \mathbb{N}\).

We require that an LRPE satisfies linearity, \(\alpha\)-reconstruction, \(\alpha\)-privacy, \(\alpha\)-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 \((x, y) \in X \times Y\) and \(z \in \mathbb{Z}_q^n\), the functions \(\text{sE}(x, \cdot), \text{mE}(z, \cdot), \text{mkE}(z, \cdot), \text{rE}(y, z, \cdot), \text{rKE}(y, z, \cdot), \text{sD}(x, y, z, \cdot), \text{dD}(x, y, z, \cdot)\) are \(\mathbb{Z}_q\)-linear.

- **(delegable):** For all \(\alpha \in \mathbb{Z}_q, z, z' \in \mathbb{Z}_q^n, w \in \mathbb{Z}_q^n\) and \(y \in Y\), there exists a linear map \(D(y, \cdot) : \text{mkE}(z, x) + \text{mE}(z, w) \mapsto \text{rKE}(y, z, \alpha) + \text{rE}(y, z, w)\).

- **(re-randomizable):** For all \(\alpha \in \mathbb{Z}_q, z, z' \in \mathbb{Z}_q^n, w \in \mathbb{Z}_q^n\), there exists a linear map \(R(z, z', \cdot) : \text{mkE}(z, x) + \text{mE}(z, w) \mapsto \text{rKE}(z', \alpha) + \text{rE}(z', w)\).

### 4.3 Construction: LR-CCA\(_{\text{C2}}\)-secure ABKEM Scheme

Here, we will give the construction of an LR-CCA\(_{\text{C2}}\)-secure ABKEM scheme for a predicate \(P\) based on our QA-NIZK argument in Section 3 and an LRPE for \(P\).

**Construction.** Let
\[
\text{L}^\text{ABE} := \left\{ (|c|_1, L) \mid \exists s \in \mathbb{Z}_q^k \text{ s.t. } c = L((A'_i)_{i \in [1, t]}) s \right\},
\]
where \(\rho := (|A|_1, (|A'|_1)_{i \in [1, t]}) \in \mathbb{G}^{(k+1) \times k} \times \mathbb{G}_1^{(k+1) \times k} \), \(k \geq 1\) is determined by the underlying assumption, and \(L\) is a \(\mathbb{Z}_q\)-linear map. Let \(\Pi = (\text{Gen, Prove, Ver, Sim})\) be an OT-SS QA-NIZK argument for \(\mathcal{L} := \left\{ L^\text{ABE} \right\}\). Let \((\text{se, mE, mkE, rE, rKE, sD, dD})\) be an LRPE for \(P\). Our ABKEM is \((\text{Setup, KGen, Encap, Decap, UpdateMSK, UpdateSK})\) is defined in Section 7.

If we instantiate ABKEM with our QA-NIZK argument \(\Pi_{\text{OT-SS}}\) in Section 3.2 that is secure under the \(\mathcal{D}_1\)-MDDH assumption, then its ciphertext is only 2 group elements larger 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 \(\Pi\). Next, we
for boolean formulae and arithmetic formulae, and broadcast
Zhang et al. [11] proposed LRPE schemes that correspond to

Remark 3 (On the concrete instantiations and leakage bounds):

Fig. 7 is derived from the parameters of the underlying LRPE.

KGen(mpk, sk):

K := Gen(par, ρ)

mpk := (|A⟩, [B], |W|, |A⟩, [W, B]|_{i∈[1,n]}, |k⟩|A⟩, r, csr)

Return (mpk, sk)

Fig. 7 Our LR-CCA2-secure ABKEM scheme ABKEM. The boxes

show the security of our ABKEM.

Theorem 2: If the Dk-MDDH problem in 𝓓 is hard and Π is an
OT-SS QA-NIZK argument, then ABKEM defined in

is (ℓmask, ℓsk)-LR-CCA2-secure, where ℓmask and ℓsk

FIGURE 7

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:

ℓmask ≤ (nζ − 1) log q + log(1 − 1/q) + 2 − ω(log λ),

ℓsk ≤ (nζ − 1) log q + log(1 − 1/q) + 2 − ω(log λ),

where n ≥ 1 is an arbitrary integer and k is determined by

the underlying assumption. (Larger nζ guarantees higher

leakage resilience, but requires longer keys.) By instanti-

Proceeding 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

ciphertexts and secret keys that will be used. Let a+ ∈ ker(A)

and b+ ∈ ker(B). A ciphertext under an attribute x ∈ 𝓋 has

the following forms:

(Normal): A normal ciphertext is generated as in the actual

scheme.

(Semi-Functional): A semi-functional (SF) ciphertext is the

same as normal ciphertext except that As is replaced by

As + b+ s, where s ← Zq. That is,

\[ \text{ct}_x = \left( \text{[A}_x \text{s]}_1 + \text{b}^{\perp}_x \text{s}, \text{e} \left( x, \text{[W}^T_i \text{b}^{\perp}_x \text{s})_{i\in[1,n]} \right) \right)_1, \pi \]

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

A secret key for an attribute \( y \in 𝓋 \) 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 \( B(r + r') \) is replaced by

\( B(r + r') + a^r \hat{r} \), where \( \hat{r} \leftarrow \mathcal{Z}_q \).

(Pseudo-SF): A pseudo-SF secret key is the same as

pseudo-normal secret key except that \( k \) is replaced by

\( k + a^\perp x \), where \( x \leftarrow \mathcal{Z}_q \). (SF): An SF secret key is the same as pseudo-sf secret key

except that \( B(r + r') + a^r \hat{r} \) is replaced by \( B(r + r') \).

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

\( [k^\perp \text{As}]_T \) for some \( s \in \mathcal{Z}_q \), and hence they completely hide

\( x \) 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

\[ \text{Adv}^{\text{LR-CCA}}_{\text{ABKEM}, \mathcal{A}}(\lambda) \leq \text{Adv}^{\text{MDDH}}_{\mathcal{A}, \mathcal{D}_k, \mathcal{B}}(\lambda) + 2Q \text{Adv}^{\text{MDDH}}_{\mathcal{B}, \mathcal{D}_k, \mathcal{B}''}(\lambda) \]
\[ + \text{Adv}_{\Pi, S^r}^{\text{OT-SS}}(\lambda) + O^{2-\omega(\log \lambda)}. \quad (6) \]

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

- **G₀**: This is the real security game defined in Figure 6.
- **G₁**: This game is the same as G₀ except that ([c']₁, π', K₀') outputted by CHAL are computed as follows:
  \[ [c']₁ = \left( sE(x, (W_j^T)_{j \in \{1, n\}}) \right) \cdot \left( sE(x, (\cdot)) \right), \]
  \[ \pi' = \text{Sim}(\text{crs}, \text{td}, ([c']₁), sE(x, \cdot)), \]
  \[ K₀' = [kT]₀T, \]

where c₀' := As.

- **G₂**: This game is the same as G₁ except that the challenge ciphertext becomes SF. Namely, c₀' = As is replaced by c₀' = As + b₅.δ.

- **G₃**: This game is the same as G₂ except that DECAP returns ⊥ for A’s queries (c₅, c₆) such that c₅ is invalid.

- **G₄,i,1**: This game is the same as G₄,i-1 except that the i-th keys revealed (or leaked) to A become pseudo-normal. The game is defined for i = 1, ..., Q.

- **G₄,i,2**: This game is the same as G₄,i,1 except that the i-th key revealed (or leaked) becomes SF. The game is defined for i = 1, ..., Q. We set G₄,0,3 := G₃.

- **G₄**: This game is the same as G₄,0,3 except that K₀' := Kₙ.

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 A when A queries to CREATE. Alternatively, the game generates a secret key if the secret key has not generated when A queries REVEAL, LEAK, or DECAP, and then adds the key to the set Qₚ. We note that this makes no difference in A’s view.

In G₄, the view of A is statistically independent of the challenge bit b. Hence, we have
\[ \text{Pr}[G₄^\Pi \Rightarrow 1] = 1/2. \quad (7) \]

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 (G₀ to G₁)**: \[ \text{Pr}[\text{LR-CCA}_{\text{CMIL}}^\Pi \Rightarrow 1] = \text{Pr}[G₀^\Pi \Rightarrow 1] = \text{Pr}[G₁^\Pi \Rightarrow 1]. \]

**Proof**. G₀ is the real security game. In G₁, the change in the way generating [c']₁ and K₀' is conceptual since
\[ c' = \begin{pmatrix} A \cdot sE(x, (W_j^T)_{j \in \{1, n\}}) \\ I_{k+1} \cdot sE(x, (\cdot)) \end{pmatrix} \text{As} \]

Moreover, we simulate the QA-NIZK proof π⁺ in CHAL(x⁺) by using Π’s zero-knowledge simulator. By the perfect zero-knowledge property of Π, G₁ is identical to G₀.

**Lemma 8 (G₁ to G₂)**: There is an adversary B breaking the D₅,MDDH assumption in G₁ with \[ \text{Adv}_{G₁, D₅, S^r}^{\text{MDDH}}(\lambda) \geq | \text{Pr}[G₁^\Pi \Rightarrow 1] - \text{Pr}[G₂^\Pi \Rightarrow 1] |. \]

**Proof**. The difference between G₁ and G₂ happens when A queries DECAP with (x, c₅, (c₁, π)) such that c₁ \not\in \text{Span}(A₀₀) and \text{Ver}(\text{crs}, (c₁, ID), π) = 1. The probability of this happening is bounded by \text{Adv}_{G₁, S^r}^{\text{OT-SS}}(\lambda), and then we have \[ | \text{Pr}[G₂^\Pi \Rightarrow 1] - \text{Pr}[G₃^\Pi \Rightarrow 1] | \leq \text{Adv}_{G₁, S^r}^{\text{OT-SS}}(\lambda). \]

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 A’s view, because the DECAP rejects all invalid ciphertexts by the change in G₁. Furthermore, the DECAP does not provide additional information to A since the DECAP returns ⊥ for A’s queries (\(\cdot), (c₁, π)) such that c₁ \not\in \text{Span}(A₀₀).

**Lemma 10 (G₃ to G₄,0,3)**: \[ \text{Pr}[G₃^\Pi \Rightarrow 1] = \text{Pr}[G₄,0,3^\Pi \Rightarrow 1]. \]

**Lemma 11 (G₄,i,1 to G₄,i,1)**: There is an adversary \(\text{B}'\) such that \[ \text{Adv}_{G₄,i,1}^{\text{MDDH}}(\lambda) \geq | \text{Pr}[G₄,i,1^\Pi \Rightarrow 1] - \text{Pr}[G₄,i,1^\Pi \Rightarrow 1] |. \]

**Lemma 12 (G₄,i,1 to G₄,i,2)**: We have \[ | \text{Pr}[G₄,i,1^\Pi \Rightarrow 1] - \text{Pr}[G₄,i,2^\Pi \Rightarrow 1] | \leq O^{2-\omega(\log \lambda)}, \] as long as the leakage amount of \(mₕ\) and \(sₕ\) are at most \(\ellₕₕ\) and \(\ellₕₕ\) bits, respectively. Here, \(\ellₕₕ\) and \(\ellₕₕ\) is derived from the parameters of the underlying LRPE.

**Lemma 13 (G₄,i,2 to G₄,i,3)**: There is an adversary \(\text{B}''\) such that \[ \text{Adv}_{G₄,i,2}^{\text{MDDH}}(\lambda) \geq | \text{Pr}[G₄,i,2^\Pi \Rightarrow 1] - \text{Pr}[G₄,i,3^\Pi \Rightarrow 1] |. \]

**Lemma 14 (G₄,Q,3 to G₃)**: \[ \text{Pr}[G₄,Q,3^\Pi \Rightarrow 1] = \text{Pr}[G₃^\Pi \Rightarrow 1]. \]

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 \(D₅\)-MDDH problem in \(G₁\) and the \(Dₖ\)-KerMDH problem in \(G₂\) are hard and \(H\) is a CR hash function family, then our ABKEM is \((\ellₕₖ, \ellₕₖ)\)-LR-CCA2 secure, where \(\ellₕₖ\) and \(\ellₕₖ\) 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., \(P(\text{ID}, \text{ID}') = 1\) if and only if \(\text{ID} = \text{ID}'\). 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 \(\ell_{sk} = \ell_{sk}(\lambda)\) be a leakage bound for secret keys. An IBKEM scheme \(IBKEM\) is LR-CCA2-secure if for any PPT adversary \(\mathcal{A}\), \(\text{Adv}^{\text{LR-CCA}_2}_{\text{IBKEM}, \mathcal{A}}(\lambda) := \Pr[\text{LR-CCA}_{\text{BML}} = 1] - \frac{1}{2}\) is negligible, where Game LR-CCA_{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 \(\text{ID}^*\) before seeing any parameters. We denote the corresponding advantage function by \(\text{Adv}^{\text{LR-CCA}_2}_{\text{IBKEM}, \mathcal{A}}(4)\).

We define leakage rate \(\gamma\) of the scheme to be value of \(\ell_{sk}/|\text{sk}_{ID}|\). 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 \(I := \mathbb{Z}_q\) be an identity space. Let

\[
\mathcal{L}^{\text{IBE}}_{\ell_{sk}} := \left\{ ([c], \text{ID}) \in \mathbb{G}_1^{2k+\mu} \times \mathbb{Z}_q^k \mid \exists s \in \mathbb{Z}_q^k \text{ s.t.} \right\}
\]

where \(\rho := ([A],[B],[B']_1) \in \mathbb{G}_1^{(k+n) \times (k+n)} \times (\mathbb{G}_1^k)^2, k \geq 2\) is determined by the underlying assumption, and \(\mu \geq 1\) is an arbitrary integer. Note that \(\mathcal{L}^{\text{IBE}}_{\ell_{sk}}\) is a special case of \(\mathcal{L}^{\text{GLS}}_p\) where \(m = 2\) and \(x_1\) is fixed to 1. As \(\mu\) increases, the efficiency of the scheme becomes less efficient, but the leakage rate of the scheme becomes greater. Let \(\Pi = (\text{Gen, Prov, Ver, Sim})\) be an OT-SS QA-NIZK argument for \(\mathcal{L} := \{\mathcal{L}^{\text{IBE}}_{\ell_{sk}}\}\). Our IBKEM = (Setup, KGen, Encap, 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 \(\Pi_{\text{OT-SS}}\) 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 \(\Pi\). Next, we give the security of our IBKEM.

**Theorem 3:** If \(\mathcal{D}_{\ell_{sk}}^{\text{exMDDH}}\) problem in \(\mathbb{G}_1\) is hard, \(\Pi\) is an OT-SS QA-NIZK argument, and

\[
\ell_{sk} \leq (\mu + k - 1) \log q - \omega(\log \lambda),
\]

then IBKEM defined in Figure 9 is \(\ell_{sk}\)-LR-CCA2-secure.

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

\[
\gamma = \frac{\ell_{sk}}{|\text{sk}_{ID}|} = \frac{(\mu + k - 1) \log q - \omega(\log \lambda)}{(\mu + 2k) \log q}
\]

**Fig. 8** Security game LR-CCA_{BML}.
If $\mu = \omega(\log \lambda)$, with $k$ fixed, the leakage rate $\gamma$ 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 $2\lambda$ as $\omega(\log \lambda)$ for $\lambda$-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

$$\text{Adv}_{\text{IBEKEM-S}}^{\text{LR-CCA2}}(\mathcal{A}) \leq \text{Adv}_{\text{IBKEM}}^\text{MMDDH}(\mathcal{B}) + \text{Adv}_{\text{IBKEM}}^\text{OT-SS}(\mathcal{B}') + 1/q + 2^{-\omega(\log \lambda)}.$$  \hspace{1cm} (9)

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

- **G1:** This game is the same as $G_0$ except that $\pi'$ in the challenge ciphertext is computed via $\text{Sim}(\text{crs}, \text{td}, ([c'], \text{ID}'))$ instead of $\text{Prove}(\text{crs}, ([c'], \text{ID}'), s)$.
- **G2:** This game is the same as $G_1$ except for the following changes:
  - $[B]_1$ and $[k]_1$ in the master public key are computed as follows:
    $$[B]_1 := [R'A - \text{ID}^* \cdot I_k]_1$$
    $$[k]_1 := \left(\begin{bmatrix} A \\ R \end{bmatrix} \right)^\top,$$
  where $R' \leftarrow Z_q^{k \times (k + \mu)}$ and $t' \leftarrow Z_q^{2k + \mu}$. In the following games, the game $s$-LR-CCA2 sets $[t']_2$ as a secret key for the challenge identity $\text{ID}^*$. Then, we have
    $$A_{\text{ID}} = \begin{bmatrix} A \\ B + \text{ID} \cdot I_k \end{bmatrix} = \begin{bmatrix} A \\ R \end{bmatrix} + (\text{ID} - \text{ID}^*) \cdot I_k.$$
- $K_{\text{ID}}^*$ outputted by CHAL is computed by $[c'] \cdot t'$. $\text{sk}_{\text{ID}}$ for $\text{ID} \neq \text{ID}^*$ is generated as follows:
  - $t_0 \leftarrow Z_q^{k + \mu}$, $[t_1']_2 := \left(\begin{bmatrix} -A^\top t_0' + k \\ \text{ID} - \text{ID}^* \end{bmatrix} \right)_2$, $\text{sk}_{\text{ID}} = [t_2']_2 := [t_0' - R^\top t_1']_2$.

We note that these values can be generated without knowing $A$, while with knowing $[A]_1$ and $[A]_2$.

- **G3:** This game is the same as $G_2$ except that $\text{DECAP}$ returns $\bot$ for $\mathcal{A}'$’s queries (ID, ct = ([c], $\pi'$)) such that $c \notin \text{Span}(A_{\text{ID}})$.
- **G4:** This game is the same as $G_3$ except that $K_{\text{ID}}^*$ outputted by CHAL is sampled from $\mathcal{G}_T$ uniformly at random.

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

$$\Pr[G_5^\text{\ll}} = 1] = 1/2.$$  \hspace{1cm} (10)

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

**Lemma 15** ($G_0$ to $G_1$): $\Pr[L_{\text{LR-CCA2}}^\text{\ll}} = 1] = \Pr[G_0^\text{\ll}} = 1] = \Pr[G_1^\text{\ll}} = 1]$.  

**Proof.** $G_0$ is the real security game. In $G_1$, we simulate the QA-NIZK proof $\pi'$ in CHAL(ID') by using $\Pi$’s zero-knowledge simulator. By the perfect zero-knowledge property of $\Pi$, we have $\Pr[G_0^\text{\ll}} = 1] = \Pr[G_1^\text{\ll}} = 1]$. \hfill $\square$

**Lemma 16** ($G_1$ to $G_2$): $\Pr[G_1^\text{\ll}} = 1] = \Pr[G_2^\text{\ll}} = 1]$.  

**Proof.** The distributions of all values is not affected by the
Lemma 17 (G2 to G3): There is an adversary $B$ that solves the $D_{k+p,k}$-exMDDH problem in $G_2$ with $\text{Adv}_\text{exMDDH}^{\text{G}_2}(\lambda) \geq |\Pr[G_2^A \Rightarrow 1] - \Pr[G_3^A \Rightarrow 1]| - 1/q$.

Proof. Using $A$, we can construct $B$ that solves the $D_{k+p,k}$-exMDDH problem in $G_1$ as follows. Let $(g, [A]_1, [A]_2, [u]_1)$ be an instance of $D_{k+p,k}$-exMDDH problem, where either $u = Aw$ for $w \leftarrow Z_q^k$ or $u = s \cdot Z_q^{k+1}$, and let $ID^*$ be a challenge identity chosen by $A$. Clearly, $B$ can generate $mpk$ and $sk_{\text{ID}}$ using $[A]_1$ and $[A]_2$, respectively. Hence, $B$ can simulate INIT, REVEAL, LEAK, and DECAP. We only focus on simulating CHAL by $B$.

When $A$ queries to CHAL, $B$ simulates the values output by CHAL as follows:

\[ [c^*]_1 := \left[ \begin{array}{c} u \\ R' u \\ \end{array} \right] \; , \; \pi := \text{Sim} (\text{crs}, \text{td}, ([c^*]_1, ID^*)) \],

\[ K^*_0 := [c^* \cdot t^*]_T, \; K^*_1 := s \cdot G_T. \]

Then, $B$ sends $([c^*]_1, \pi, K^*_0)$ to $A$. To conclude this proof, we show that the distribution of $c^*$ is the same as one in $G_2$ if $u = Aw$ and in $G_3$ otherwise.

- Case $u = Aw$: In this case, we have
  \[ c^* := \left[ \begin{array}{c} u \\ R' u \\ \end{array} \right] \; = \; \left[ \begin{array}{c} A \\ R' A \\ \end{array} \right] \; w = A_{ID} \cdot w. \]

Hence, the distribution of $c^*$ is the same as one in $G_2$.

- Case $u \leftarrow Z_q^{k+p}$: It is sufficient to show that $R' u$ is uniformly distributed over $Z_q^{k+p}$ from $A$'s viewpoint. With probability $1 - 1/q$, $u$ is linearly independent of $A$ since $A$ is full rank and $u$ is uniformly distributed over $Z_q^{k+p}$. Hence, $R' u$ is uniformly distributed over $Z_q^{k+1}$ even given $A$, $R' A$, and $u$, which are all the information that $A$ knows. Therefore, the distribution of $c^*$ is the same as one in $G_3$.

From these, we have $|\Pr[G_2^A \Rightarrow 1] - \Pr[G_3^A \Rightarrow 1]| \leq \text{Adv}_\text{exMDDH}^{\text{G}_2}(\lambda) + 1/q$. □

Lemma 18 (G3 to G4): There is an adversary $B'$ breaking the OT-SS of $\Pi$ with $\text{Adv}_D^{\text{OT-SS}}(\lambda) \geq |\Pr[G_3^A \Rightarrow 1] - \Pr[G_4^A \Rightarrow 1]|$.

Proof. The difference between $G_3$ and $G_4$ happens when an adversary queries DECAP with $(ID, t = ([c]_1, \pi))$ such that $c \notin \text{Span}(A_{ID})$ and $\text{Ver}([c]_1, (c_1, ID), \pi) = 1$. The probability of this happening is bounded by $\text{Adv}_D^{\text{OT-SS}}(\lambda)$, and then we have $|\Pr[G_3^A \Rightarrow 1] - \Pr[G_4^A \Rightarrow 1]| \leq \text{Adv}_D^{\text{OT-SS}}(\lambda)$. □

By the change in $G_4$, in the following game, $A$ learns information about $t'$ only from $k$ in $mpk$ and $\{f(s_{\text{sk}_{ID}})\}$ which is $\ell_{sk}$-bit leakage of $sk_{ID}$ since DECAP returns $\bot$ for $A$'s queries (ID, ([c]_1, \pi)) such that $c \notin \text{Span}(A_{ID})$.

Lemma 19 (G4 to G3): $|\Pr[G_3^A \Rightarrow 1] - \Pr[G_4^A \Rightarrow 1]| \leq 2^{-\omega(\log 4)}$.

To prove this lemma, we use the following lemma.

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

\[ \Delta \left((H, H(X), f(X)), (H, U_y, f(X))\right) \leq \frac{1}{2} \sqrt{\gamma(X) \cdot |Y| \cdot |Z|}, \]

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

Proof of Lemma 19. To bound this, we show that the two distributions $(c^*, c^* \cdot t^*, k, \{f(s_{sk_{ID}})\})$ and $(c^*, u, k, \{f(s_{sk_{ID}})\})$ are statistically indistinguishable, where $u \leftarrow Z_q$. We can see that $H_{t^*} := c^* \cdot t^*$ is a universal hash function. By Lemma 20 and Eq. (8), we have

\[ \Delta \left((c^*, c^* \cdot t^*, k, \{f(s_{sk_{ID}})\}), (c^*, u, k, \{f(s_{sk_{ID}})\})\right) \leq \frac{1}{2} \sqrt{\gamma \cdot (k+p) \cdot q \cdot (q^k \cdot 2^\omega)} \]

\[ = \frac{1}{2} \sqrt{2^{-k-1} \log q \cdot 2^\omega} \]

\[ = 2^{-\omega(\log 4)}. \]

Hence, we have $|\Pr[G_3^A \Rightarrow 1] - \Pr[G_4^A \Rightarrow 1]| \leq 2^{-\omega(\log 4)}$. □

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 $D_{k+p,k}$-exMDDH problem in $G_1$ and the $D_{k'}$-KerMDDH problem in $G_2$ are hard, $H$ is a CR hash function family, and Eq. (8) holds, then our IBKEM is $\ell_{sk}$-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 $I := \{0,1\}^m$ be an identity space. Our LR-CCA2-secure IBKEM scheme is obtained by changing the scheme in previous section as follows:

- We set $mpk := ([A]_1, ([B]_1)_i \in \{0,1\}^m, \{k\}_1, \text{crs})$ and $sk_{ID} := ([A]_1, ([B]_1)_i \in \{0,1\}^m, \{k\}_1)$, where $B_0, \ldots, B_m \in Z_q^{k \times k}$.

- For $ID := (ID_1, \ldots, ID_m) \in \{0,1\}^m$, we set
- We use a QA-NIZK argument for a language
  \[ L_{\rho}^{\text{IBE}} := \{ (|e|_1, \mathbf{ID}) \mid |s| \in \mathbb{Z}_q^{k \cdot s.t. \mathbf{c} = \mathbf{A}_s \mathbf{d}_s \} , \]
  where \( \rho := (|A|_1, \{|B|_1|_1\}_{i \in [0, m]}) \).

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


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


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

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

**Theorem 4:** $\Pi_{\text{USS}}$ defined in Figure A-1 has perfect completeness and perfect zero-knowledge. Furthermore, if the $\mathcal{D}_K$-MDDH problem in $G_1$ and the $\mathcal{D}_K$-KerMDDH problem in $G_2$ is hard and $\mathcal{H}$ is a CR hash function family, then $\Pi_{\text{USS}}$ 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 $c \in M_r$, we have $P^c_r = \sum_{i=1}^m x_i P^i_r = K^i c M_r = K^i c$ and for all $A$, $B$, $K^i_r$, and $K^i$, we have $[w^T B^T (K^i_r + \tau K^i)]_1 \circ [A]_2 = [w^T B^T]_1 \circ [K^i_r A + \tau K^i A]_2$.

**Fig. A-1** Our USS QA-NIZK argument $\Pi_{\text{USS}}$. where $\tau = H([c]_1, [t]_1)$.

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