SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Provably Secure Attribute-Based Authenticated Encryption with Keyword Search from Ideal Lattices

Lisha YAO[†1], Lin HOU[†1], Jian WENG[1*] & Feixiang ZHAO[1]

[1]*College of Cyber Security, Jinan University, Guangzhou* 510632*, China*

Appendix A Related Work

In PAEKS, the ciphertext is authenticated using the sender's secret key, which resists insider keyword guessing attacks (IKGA). Many PAEKS schemes have been proposed in the literature with various attractive functions, such as certificateless [8, 12], designated-server [10], et al. Liu et al. [15] recently proposed a quantum-resistant PAEKS scheme. Its security model was further enhanced to include multi-ciphertext and multi-trapdoor [16].

In multi-user scenarios, attribute-based keyword search (ABKS) is one of the primitives used to implement keyword search effectively. Various ABKS schemes are proposed in [14, 19, 20, 23, 24]. These constructions, however, have a fragile level of security when faced with quantum computers. Li et al. [11] designed the first ABKS scheme from lattices with fine-grained control of the searchability against quantum attacks. Although their construction could resist IKGA, their syntax, security model and underlying assumptions differ from that of AB-AEKS. This scheme can be used as a building block to construct our AB-AEKS scheme. By integrating the search function into an attribute-based signcryption, Liu and Fan [13] proposed the concept of searchable attribute-based authenticated encryption. However, their instantiation cannot withstand quantum cryptanalysis.

Appendix B Security Model for AB-AEKS

We propose two security models that resist two kinds of attacks: chosen-keyword attacks (CKA) and insider keyword guessing attacks (IKGA), focusing on properties that are ciphertext indistinguishable and ciphertext unforgeable even for malicious cloud servers. They are illustrated as an interactive process between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ as follows:

Definition 1 (Security Game for IND-CKA)**.** We describe the security model of indistinguishable under chosen-keyword attacks (IND-CKA) in the following way.

Setup. The challenger runs Setup algorithm to compute a master key pair (MPK, MSK) and returns MPK to adversary.

Query phase 1. The adversary issues the following queries:

- $\mathcal{O}_{\mathsf{KeyGen}_s}$: $\mathcal{A}$ could query the sender's key pair using an identity ID. $\mathcal{C}$ computes $(PK_s, SK_s) \leftarrow \mathsf{KeyGen}_s(MPK, MSK, ID)$ and returns (PK_s, SK_s).
- $\mathcal{O}_{\mathsf{KeyGen}_r}$: $\mathcal{A}$ could query the receiver's secret key using an access policy f except for $f(\mathbf{x}^*) = 0$, where $\mathbf{x}^*$ denotes a challenge attribute vector. $\mathcal{C}$ computes $SK_r \leftarrow \mathsf{KeyGen}_r(MPK, MSK, f)$ and returns SK_r.
- $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$: $\mathcal{A}$ transmits a keyword w, an attribute vector $\mathbf{x}$, and identity ID. $\mathcal{C}$ evaluates $C \leftarrow \mathsf{AB\text{-}AEKS}(MPK, PK_s, SK_s, \mathbf{x}, w)$, where $(PK_s, SK_s) \leftarrow \mathsf{KeyGen}_s(MPK, MSK, ID)$, and returns C to $\mathcal{A}$.
- $\mathcal{O}_{\mathsf{Token}}$: $\mathcal{A}$ transmits a keyword w and an access policy f, except for $f(\mathbf{x}^*) = 0$. $\mathcal{C}$ evaluates $K \leftarrow \mathsf{Token}(MPK, SK_r, w)$, where $SK_r \leftarrow \mathsf{KeyGen}_r(MPK, MSK, f)$, and returns K to $\mathcal{A}$.

Challenge. The adversary submits two keywords w_0^*, w_1^*, an identity ID^* and a target attribute vector $\mathbf{x}^*$. $\mathcal{C}$ chooses randomly a bit $r \in \{0, 1\}$ and runs $\mathsf{AB\text{-}AEKS}$ algorithm to generate a challenge ciphertext C^* under the keyword w_r^*.

Query phase 2. $\mathcal{A}$ continues to query similar to phase 1.

Guess. The adversary outputs a guess $r' \in \{0, 1\}$.

$\mathcal{A}$ wins the game if $r' = r$. We say that the AB-AEKS scheme is secure if for all probabilistic polynomial time (PPT) adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ wins in the game is negligible.

Definition 2 (Security Game for UNF-IKGA)**.** We describe the security model of unforgeable under insider keyword guessing attacks (UNF-IKGA) in the following way.

Setup. Same as Setup in the IND-CKA game.

Query phase. The adversary issues the following queries:

- $\mathcal{O}_{\mathsf{KeyGen}_s}, \mathcal{O}_{\mathsf{KeyGen}_r}$: Same as $\mathcal{O}_{\mathsf{KeyGen}_s}$ and $\mathcal{O}_{\mathsf{KeyGen}_r}$ in the IND-CKA game, respectively.
- $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}, \mathcal{O}_{\mathsf{Token}}$: Same as $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$ and $\mathcal{O}_{\mathsf{Token}}$ in the IND-CKA game, respectively.

Forgery. The adversary outputs a forgery C^* associated with $(w^*, \mathbf{x}^*, ID^*)$, subject to the restriction that ID^* cannot be queried in $\mathcal{O}_{\mathsf{KeyGen}_s}$ and $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$.

$\mathcal{A}$ wins the game if the forgery passes test algorithm. We say that the AB-AEKS scheme is secure if for all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ wins in the game is negligible.

Selective-attribute security. If the adversary $\mathcal{A}$ has to initiate the target attribute vector $\mathbf{x}^*$ in the Setup phase before being given MPK in the above security games, we call it selective-attribute security.

* Corresponding author (email: cryptjweng@gmail.com)

†) Lisha Yao and Lin Hou are co-first authors of the article.

Appendix C Proof of Theorem 1

Before proving Theorem 1, we give the definitions of the three primitives involved in the general construction, including digital signature, attribute-based keyword search, and non-interactive zero-knowledge proof.

Appendix C.1 Digital Signature

Syntax. A signature scheme with message space $\mathcal{M}$ consists of the following PPT algorithms:

- $\mathsf{KeyGen}(1^\lambda) \to (vk, sk)$: Take as input the security parameter λ, output a verification key vk and a signing key sk.
- $\mathsf{Sign}(sk, m) \to \sigma$: Take as input the signing key sk and message $m \in \mathcal{M}$, output a signature σ.
- $\mathsf{Ver}(vk, (m, \sigma)) \to 0/1$: Take as input the verification key vk, message $m \in \mathcal{M}$, and signature σ, output 1 if σ is a valid signature for m. Otherwise, output 0.

Correctness. The signature scheme with message space $\mathcal{M}$ is correct: for all $\lambda \in \mathbb{N}$ and $m \in \mathcal{M}$, let $(vk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda)$, if there exists $\sigma \leftarrow \mathsf{Sign}(sk, m)$, then we have $1 \leftarrow \mathsf{Ver}(vk, (m, \sigma))$.

EUF-CMA. The existentially unforgeable under chosen-message attacks (EUF-CMA) security game is defined between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ as below.

- **Setup**. $\mathcal{C}$ generates $(vk, sk) \leftarrow \mathsf{Setup}(1^\lambda)$ and gives vk to $\mathcal{A}$.
- **Queries**. For a signing query m, $\mathcal{C}$ returns $\sigma \leftarrow \mathsf{Sign}(sk, m)$ to $\mathcal{A}$.
- **Forgery**. $\mathcal{A}$ outputs a forgery (m^*, σ^*).

$\mathcal{A}$ wins the game if $\mathcal{C}$ captures $\mathsf{Ver}(vk, (m^*, \sigma^*)) \to 1$, subject to the restriction that of m^* cannot be queried. We say that the signature scheme is secure if for all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ wins in the game is negligible.

Appendix C.2 Attribute-Based Keyword Search

Syntax. For a keyword space $\mathcal{W}$ and policy space $\mathcal{F} : \{0,1\}^l \to \{0,1\}$, a (key-policy) attribute-based keyword search (ABKS) scheme is made of the following PPT algorithms:

- $\mathsf{Setup}(1^\lambda, 1^l) \to (mpk, msk)$: Take as input the security parameter λ and the number of attributes l, output a master public key mpk and a master secret key msk.
- $\mathsf{KeyGen}(mpk, msk, f) \to sk_f$: Take as input the master public key mpk, master secret key msk, and an access policy $f \in \mathcal{F}$, output a secret key sk_f.
- $\mathsf{ABKS}\,(mpk, \mathbf{x}, w) \to c$: Take as input the master public key mpk, attribute $\mathbf{x} \in \{0,1\}^l$ and a keyword $w \in \mathcal{W}$, output a ciphertext c.
- $\mathsf{Token}\,(mpk, sk_f, w) \to k$: Take as input the master public key mpk, secret key sk_f and a keyword $w \in \mathcal{W}$, output a keyword token k.
- $\mathsf{Test}\,(c, k) \to \{0,1\}$: Take as input the ciphertext c and token k, output 1 if $f(\mathbf{x}) = 0$ and c and k contain the same keyword; otherwise, output 0.

Correctness. The ABKS scheme is correct: For all $\lambda \in \mathbb{N}$, keyword space $\mathcal{W}$, and policy space $\mathcal{F}$, let $(mpk, msk) \leftarrow \mathsf{Setup}(1^\lambda, 1^l)$ and $sk_f \leftarrow \mathsf{KeyGen}(mpk, msk, f)$, if there exists $c \leftarrow \mathsf{ABKS}\,(mpk, \mathbf{x}, w)$, $k \leftarrow \mathsf{Token}\,(mpk, sk_f, w)$ and $f(\mathbf{x}) = 0$, then we have $1 \leftarrow \mathsf{Test}\,(c, k)$.

IND-CKA. The indistinguishable under chosen-keyword attacks (IND-CKA) security game is defined between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ as below.

- **Setup**. $\mathcal{C}$ generates $(mpk, msk) \leftarrow \mathsf{Setup}(1^\lambda, 1^l)$ and returns mpk to $\mathcal{A}$.
- **Query phase 1**. $\mathcal{A}$ makes the following queries:
  - $\mathcal{O}_{\mathsf{KeyGen}}$: For a key query f except for $f(\mathbf{x}^*) = 0$, where $\mathbf{x}^*$ denotes a challenge attribute vector. $\mathcal{C}$ returns $sk_f \leftarrow \mathsf{KeyGen}(mpk, msk, f)$ to $\mathcal{A}$.
  - $\mathcal{O}_{\mathsf{Token}}$: For a token query (f, w), except for $f(\mathbf{x}^*) = 0$. $\mathcal{C}$ computes $k \leftarrow \mathsf{Token}(mpk, sk_f, w)$, where $sk_f \leftarrow \mathsf{KeyGen}(mpk, msk, f)$, and returns k to $\mathcal{A}$.
- **Challenge**. The adversary submits a challenge tuple $(\mathbf{x}^*, w_0^*, w_1^*)$. $\mathcal{C}$ chooses randomly a bit $b \in \{0,1\}$ and returns $c^* \leftarrow \mathsf{ABKS}(mpk, \mathbf{x}^*, w_b^*)$.
- **Query phase 2**. $\mathcal{A}$ continues to queries similar to phase 1.
- **Guess**. The adversary outputs a guess $b' \in \{0,1\}$.

$\mathcal{A}$ wins the game if $b' = b$. We say that the ABKS scheme is secure if for all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ wins in the game is negligible.

Example. The ABKS scheme [11] of Li et al. satisfies the above definition and security model. For a token query (f, w), if $f(\mathbf{x}^*) \neq 0$, the challenger invokes the KeyGen algorithm to generate the token. Otherwise, $f(\mathbf{x}^*) = 0 \wedge w \neq w^*$, the challenger computes $\mathbf{T}_{f,w}$ by using $\mathbf{G}$-trapdoor and then captures the token in a normal way. For more details, please refer to [11].

Appendix C.3 Non-Interactive Zero-Knowledge Proof

Syntax. Let R be a relation corresponding to an NP language L. A non-interactive zero-knowledge (NIZK) proof system [2] contains the following PPT algorithms:

- $\mathsf{I}(1^\lambda) \to \omega$: Take as input the security parameter λ, output a common reference string ω.
- $\mathsf{P}(\omega, (y, x)) \to \pi$: Take as input the common reference string ω and an NP relation $(y, x) \in R$, output a proof π.
- $\mathsf{V}(\omega, (y, \pi)) \to 0/1$: Take as input the common reference string ω, an instance y and a proof π, output 0 or 1.

Correctness. The NIZK proof system for relation R is correct: for all $\lambda \in \mathbb{N}$ and $(y, x) \in R$, if there exists $\omega \leftarrow \mathsf{I}(1^\lambda)$ and $\pi \leftarrow \mathsf{P}(\omega, (y, x))$, then we have $1 \leftarrow \mathsf{V}(\omega, (y, \pi))$.

The NIZK proof system have two useful properties. One is called adaptive multi-theorem zero-knowledge, means that a proof is generated honestly does not leak any information beyond the fact that $y \in L$. The other is called knowledge soundness, requires that if an adversary could generate a valid proof for some statements, then it must know the corresponding witness.

Adaptive Multi-Theorem Zero-Knowledge. A NIZK proof system for relation R satisfies adaptive multi-theorem zero-knowledge if there exists a PPT simulator $\mathsf{Z} = (\mathsf{Z}_0, \mathsf{Z}_1)$ such that the following holds:

- Algorithm Z_0 outputs ω and a simulation trapdoor ζ.
- For all PPT distinguishers $\mathcal{D}$, we have

$$|\Pr[\mathcal{D}^{\mathsf{P}(\omega,(\cdot,\cdot))}(\omega) = 1 : \omega \leftarrow \mathsf{I}(1^\lambda)] - \Pr[\mathcal{D}^{\mathsf{O}(\zeta,(\cdot,\cdot))}(\omega) = 1 : (\omega, \zeta) \leftarrow \mathsf{Z}_0(1^\lambda)]| \leqslant \mathsf{negl}(\lambda),$$

where the oracle $\mathsf{O}(\zeta, (\cdot, \cdot))$ takes as input ζ and a pair (y, x), returns $\mathsf{Z}_1(\zeta, y)$ if $(y, x) \in R$, and returns $\perp$ otherwise.

Knowledge Soundness. A NIZK proof system for relation R satisfies knowledge soundness if there exists a PPT extractor $\mathsf{K} = (\mathsf{K}_0, \mathsf{K}_1)$ such that the following holds:

- Algorithm K_0 outputs ω and an extraction trapdoor ξ, where the distribution of ω is computationally close to the output distribution of $\mathsf{I}(1^\lambda)$.
- For all PPT adversaries $\mathcal{A}$, we have

$$\Pr\left[\begin{array}{l} \\ \mathsf{V}(\omega, (y, \pi)) \to 1 \wedge (y, x) \notin R : \\ \\ \end{array} \begin{array}{l} (\omega, \xi) \leftarrow \mathsf{K}_0(1^\lambda) \\ (y, \pi) \leftarrow \mathcal{A}(\omega) \\ x \leftarrow \mathsf{K}_1(\xi, (y, \pi)) \end{array}\right] \leqslant \mathsf{negl}(\lambda).$$

Appendix C.4 Security Proof for the general AB-AEKS

Theorem 1 contains the scheme's two security requirements; for clarity, we argue for the scheme's security through the following two lemmas.

Lemma 1. If Π_1 satisfies IND-CKA security and Π_3 satisfies adaptive multi-theorem zero knowledge, then the AB-AEKS scheme Π satisfies IND-CKA security.

Proof. We use hybird argument. Consider the following games:

Game 0: Identical to the IND-CKA game of Π. Suppose there exists an adversary $\mathcal{A}$ to break IND-CKA security of the AB-AEKS scheme Π with non-negligible advantage, then we build an algorithm $\mathcal{C}$ that wins IND-CKA game of ABKS with the same advantage.

1. $\mathcal{C}$ receives mpk. It then returns $MPK = (mpk, vk, \omega)$ to $\mathcal{A}$, where $(vk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda)$ and $\omega \leftarrow \mathsf{I}(1^\lambda)$.
2. $\mathcal{C}$ answers $\mathcal{A}$'s queries in the following way:
   - $\mathcal{O}_{\mathsf{KeyGen}_s}$: Upon input ID, compute $\sigma \leftarrow \mathsf{Sign}(sk, ID)$ and return $(PK_s, SK_s) = (ID, \sigma)$.
   - $\mathcal{O}_{\mathsf{KeyGen}_r}$: Upon input f, capture sk' by sending f to key generation oracle $\mathcal{O}_{\mathsf{KeyGen}'}$ of ABKS. It returns $SK_r = sk'$.
   - $\mathcal{O}_{\mathsf{AB-AEKS}}$: Upon input $(\mathbf{x}, w, ID)$, compute $c \leftarrow \mathsf{ABKS}'(mpk, \mathbf{x}, w)$ and $\pi \leftarrow \mathsf{P}(\omega, (mpk, vk, c), (ID, \sigma))$, where $\sigma \leftarrow \mathsf{Sign}(sk, ID)$. It returns $C = (c, \pi)$.
   - $\mathcal{O}_{\mathsf{Token}}$: Upon input (f, w), capture k' by sending (f, w) to token generation oracle $\mathcal{O}_{\mathsf{Token}'}$ of ABKS. It returns $K = k'$.
3. Receive $(\mathbf{x}^*, w_0^*, w_1^*, ID^*)$. $\mathcal{C}$ sends $(\mathbf{x}^*, w_0^*, w_1^*)$ to the challenger of ABKS.
4. Capture c^*, let $\pi^* \leftarrow \mathsf{P}(\omega, (mpk, vk, c^*), (ID^*, \sigma^*))$, where $\sigma^* \leftarrow \mathsf{Sign}(sk, ID^*)$. It returns $C^* = (c^*, \pi^*)$.
5. Answer the subsequent queries as in step 2.
6. $\mathcal{C}$ outputs $\mathcal{A}$'s guess as the answer to IND-CKA challenge of ABKS it is trying to solve.

Hence, we conclude that if $\mathcal{A}$ breaks the security of AB-AEKS with non-negligible advantage, then $\mathcal{C}$ wins the IND-CKA game with non-negligible advantage under the ABKS scheme.

Game 1: Identical to Game 0 except that change the way the proof π generated. Concretely, the challenger computes $(\omega, \zeta) \leftarrow \mathsf{Z}_0(1^\lambda)$ in the Setup phase. When the adversary makes encryption and challenge queries, the challenger generates $\pi \leftarrow \mathsf{Z}_1(\zeta, (mpk, pk, c))$, where c is produced by ABKS' algorithm of the ABKS scheme. By the adaptive multi-theorem zero-knowledge property of that NIZK proof system, the view of the adversary is altered only negligibly between Game 0 and Game 1.

Based on the IND-CKA security property of ABKS scheme, it shows that no adversary has non-negligible chance in wining Game 1.

Therefore, combining the above statements together, the lemma is proven. ■

Lemma 2. If Π_2 satisfies UNF-CMA security and Π_3 satisfies knowledge soundness, then the AB-AEKS scheme Π satisfies UNF-IKGA security.

Proof. Suppose there exists an adversary $\mathcal{A}$ to break UNF-IKGA security of the AB-AEKS scheme Π with non-negligible advantage, then we build an algorithm $\mathcal{C}$ that wins UNF-CMA game under the signature scheme Π_2 with the same advantage.

1. $\mathcal{C}$ receives vk. It then returns $MPK = (mpk, vk, \omega)$ to $\mathcal{A}$, where $(mpk, msk) \leftarrow \mathsf{Setup}'(1^\lambda)$ and $(\omega, \xi) \leftarrow \mathsf{K}_0(1^\lambda)$.
2. $\mathcal{C}$ answers $\mathcal{A}$'s queries in the following way:
   - $\mathcal{O}_{\mathsf{KeyGen}_s}$: Upon input ID, capture σ by sending ID to signing oracle $\mathcal{O}_{\mathsf{Sign}}$ of the signature scheme. It returns $(PK_s, SK_s) = (ID, \sigma)$.
   - $\mathcal{O}_{\mathsf{KeyGen}_r}$: Upon input f, compute $sk' \leftarrow \mathsf{KeyGen}'(mpk, msk, f)$ and return $SK_r = sk'$.
   - $\mathcal{O}_{\mathsf{AB-AEKS}}$: Upon input $(\mathbf{x}, w, ID)$, capture σ by giving ID to signing oracle $\mathcal{O}_{\mathsf{Sign}}$ of the signature scheme, evaluate $c \leftarrow \mathsf{ABKS}'(mpk, \mathbf{x}, w)$ and $\pi \leftarrow \mathsf{P}(\omega, (mpk, vk, c), (ID, \sigma))$. It returns $C = (c, \pi)$.
   - $\mathcal{O}_{\mathsf{Token}}$: Upon input (f, w), compute $k' \leftarrow \mathsf{Token}'(mpk, sk', w)$, where $sk' \leftarrow \mathsf{KeyGen}'(mpk, msk, f)$. It returns $K = k'$.
3. Receive $C^* = (c^*, \pi^*)$ associated with $(\mathbf{x}^*, w^*, ID^*)$ and check whether $\mathsf{V}(\omega, (mpk, vk, c^*), \pi^*) = 0$ is true. If the equation holds, abort. Otherwise, $\mathcal{C}$ extracts $(ID^*, \sigma^*) \leftarrow \mathsf{K}_1(\xi, (mpk, vk, c^*), \pi^*)$ and returns (ID^*, σ^*) as a forgery to the challenger of the signature scheme.

Based on the knowledge soundness property of NIZK proof system, we have ω that generated from $\mathsf{I}(1^\lambda)$ and $\mathsf{K}_0(1^\lambda)$ is computational indistinguishable. This means that the proof π computed by challenger in the oracle $\mathcal{O}_{\mathsf{AB-AEKS}}$ as an response and π^* returned by adversary as a forgery are valid. Therefore, we have that σ^* is a valid signature for ID^*. In other words, we conclude that if $\mathcal{A}$ breaks the security of AB-AEKS with non-negligible advantage, then $\mathcal{C}$ wins the UNF-CMA game with non-negligible advantage under the signature scheme. This completes the proof. ■

Table D1 The notation explanation

Notation	Description
$\mathcal{R}=\mathbb{Z}[x]/(x^n+1)$	cyclotomic ring, where n is a power of 2
$\mathcal{R}_q=\mathcal{R}/q\mathcal{R}$	quotient ring, where q is a prime such that $q\equiv 1 \mod 2n$
$\mathcal{R}_q^{1\times m}, \mathcal{R}_q^m, \mathcal{R}_q^{m\times m}$	row vector, column vector, and matrix in $\mathcal{R}_q$
$\|\mathbf{x}\|$	Euclidean norm (l_2 norm) of $\mathbf{x}$
$\|\mathbf{T}\|=\max_i\|\mathbf{t}_i\|$	maximum norm of its column vectors
$\mathbf{A}\xleftarrow{R}\mathcal{R}_q^{m-k}$	$\mathbf{A}$ is sampling from a uniformly random distribution $\mathcal{R}_q^{m-k}$
$\widetilde{\mathbf{R}}$	Gram-Schmidt orthogonalization of $\mathbf{R}$
$\mathbf{a}\|\mathbf{b}, \mathbf{A}\|\mathbf{B}$	horizontal concatenation of vectors or matrices
$\mathbf{a};\mathbf{b}, \mathbf{A};\mathbf{B}$	vertical concatenation of vectors or matrices

Appendix D Lattice-Based AB-AEKS

We introduce the preliminaries of lattice-based AB-AEKS construction, provide the correctness analysis for the proposed scheme, and give a comparative table of our construction and related schemes.

Appendix D.1 Preliminaries

Notation. Let boldface symbols denote vectors or matrices and regular lowercase letters denote single elements. Table D1 lists several notation explanations in our paper to provide a more intuitive understanding.

Background on Lattices. We provide an overview of the background on lattices, including discrete Gaussian, the hardness assumption of ring variants, and tailcut property.

Definition 3 (Lattice [1])**.** The lattice is a discrete additive subgroup on $\mathbb{R}^m$, which can be simply regarded as a set of points regularly arranged in an infinite space. We will use a special type of lattices: for positive integers m and q prime, let $\mathbf{A}\in\mathcal{R}^{1\times m}$ and $u\in\mathcal{R}_q$, define the following m-dimensional q-*ary* lattices:

$$\Lambda^{\perp}(\mathbf{A})=\left\{\mathbf{x}\in\mathcal{R}^m:\mathbf{A}\mathbf{x}=0 \bmod q\right\},$$

$$\Lambda_u^{\perp}(\mathbf{A})=\left\{\mathbf{x}\in\mathcal{R}^m:\mathbf{A}\mathbf{x}=u \bmod q\right\}=\Lambda^{\perp}(\mathbf{A})+\mathbf{z},$$

where $\mathbf{z}\in\Lambda_u^{\perp}(\mathbf{A})$. Hence, $\Lambda_u^{\perp}(\mathbf{A})$ is a coset of $\Lambda^{\perp}(\mathbf{A})$.

Definition 4 (Discrete Gaussian [1, 7])**.** For any $\sigma\in\mathbb{R}$ define the Gaussian function on $\mathbb{R}^n$ of center $\mathbf{c}$ and parameter σ:

$$\forall\mathbf{x}\in\mathbb{R}^n, \rho_{\mathbf{c},\sigma}(\mathbf{x})=\exp\left(-\pi\left\|\mathbf{x}-\mathbf{c}\right\|^2/\sigma^2\right).$$

The discrete Gaussian distribution of center $\mathbf{c}\in\mathbb{R}^n$ and distribution parameter $\sigma\in\mathbb{R}$ over a lattice $\Lambda\subset\mathbb{R}^n$ is

$$\forall\mathbf{x}\in\Lambda, \mathcal{D}_{\Lambda,\mathbf{c},\sigma}=\rho_{\mathbf{c},\sigma}(\mathbf{x})/\rho_{\mathbf{c},\sigma}(\Lambda),$$

where $\rho_{\mathbf{c},\sigma}(\Lambda)=\sum_{\mathbf{z}\in\Lambda}\rho_{\mathbf{c},\sigma}(\mathbf{z})$. Specifically, the Gaussian distribution $\mathcal{D}_{\mathcal{R},\sigma}$ ($\mathbf{c}=\mathbf{0}$ when ommitted) used in this paper denotes the discrete Gaussian sampling based on the cyclotomic ring $\mathcal{R}$.

Definition 5 ($Ring\text{-}LWE_{n,q,m,D_{\mathcal{R},\sigma}}$ [4, 18])**.** Given integers n, m, a prime integer q and a discrete Gaussian distribution $\mathcal{D}_{\mathcal{R},\sigma}$. The decisional ring-LWE problem is to distinguish the pair $(\mathbf{a}, \mathbf{a}s+\mathbf{e})$ from $(\mathbf{a}, \mathbf{b})$, where $\mathbf{a}\xleftarrow{R}\mathcal{R}_q^m, s\xleftarrow{R}\mathcal{R}_q, \mathbf{e}\xleftarrow{R}\mathcal{D}_{\mathcal{R}^m,\sigma}$ and $\mathbf{b}\xleftarrow{R}\mathcal{R}_q^m$.

Definition 6 ($Ring\text{-}ISIS_{q,m,\beta}$ [17,22])**.** Given a integer m, a prime integer q and a real number $\beta>0$. The ring-ISIS problem is to find a non-zero vector of the small polynomial $\mathbf{x}\in\mathcal{R}^m$ such that $\mathbf{a}^T\mathbf{x}=u \mod q$ and $0<\|\mathbf{x}\|\leqslant\beta$, where $\mathbf{a}\xleftarrow{R}\mathcal{R}_q^m$ and $u\xleftarrow{R}\mathcal{R}_q$.

Lemma 3 (Tail inequality [7])**.** For any $\epsilon>0, s\geqslant\eta_\epsilon(\mathbb{Z}), t>0$, we have

$$\Pr_{x\sim\mathcal{D}_{\mathbb{Z},s}}[|x|\geqslant t\cdot s]\leqslant 2e^{-\pi t^2}\cdot\frac{1+\epsilon}{1-\epsilon},$$

where $\eta_\epsilon(\mathbb{Z})$ and $\mathcal{D}_{\mathbb{Z},s}$ are smoothing parameter and discrete Gaussian of the lattice $\mathbb{Z}$, respectively. For $\epsilon\in(0,1/2)$ and $t\geqslant\omega(\sqrt{\log n})$, the probability that $|x|\geqslant t\cdot s$ is negligible. In this paper, a vector $\mathbf{x}$ sampled in $\mathcal{D}_{\mathcal{R}^m,s}$ would have small norm $\|\mathbf{x}\|\leqslant ts\sqrt{mn}$ with overwhelming probability.

Lattice Algorithm. In this work, we will utilize several lattice algorithms, which are shown in the following lemmas:

Lemma 4 (Trapdoor Generation [3, 21])**.** Given a vector $\overline{\mathbf{A}}\in\mathcal{R}_q^{m-k}$, tag $H\in\mathcal{R}_q$, Gaussian parameter σ, and ring modulus q. Algorithm $\mathsf{TrapGen}(\overline{\mathbf{A}}, H, \sigma, q)$ outputs a (pseudo)random vector $\mathbf{A}\in\mathcal{R}_q^{1\times m}$ and its trapdoor $\mathbf{R}\in\mathcal{R}^{(m-k)\times k}$ of the norm bounded by $t\sigma\sqrt{(m-k)n}$. Let $\mathbf{G}=\left(1,2,4,\cdots,2^k\right)\in\mathcal{R}_q^{1\times k}(k=\lceil\log_2 q\rceil)$ be a gadget vector, we have $\mathbf{A}=(\overline{\mathbf{A}}^T|H\mathbf{G}-\overline{\mathbf{A}}^T\mathbf{R})$ such that $\mathbf{A}(\mathbf{R};\mathbf{I}_k)=H\mathbf{G}$.

When the tag $H=0$, the output $(\mathbf{A},\mathbf{R})$ is also valid, where $\mathbf{A}=(\overline{\mathbf{A}}^T|-\overline{\mathbf{A}}^T\mathbf{R})$ is δ-uniform for some $\delta=\mathsf{negl}(n)$.

Lemma 5 (Gaussian preimage samping [3, 21])**.** Given a vector $\mathbf{A}\in\mathcal{R}_q^{1\times m}$ associated with an invertible tag $H\in\mathcal{R}_q$ and its trapdoor $\mathbf{R}\in\mathcal{R}^{(m-k)\times k}$, a polynomial $u\in\mathcal{R}_q$ and three Gaussian parameters ζ,α,σ. Algorithm $\mathsf{SamplePre}(\mathbf{A},\mathbf{R},H,u,\zeta,\alpha,\sigma)$ outputs a vector $\mathbf{x}$ sampled from a discrete Gaussian distribution $\mathcal{D}_{\mathcal{R}_q^m,\zeta}$ such that $\mathbf{A}\mathbf{x}=u$.

Lemma 6 (Trapdoor delegation [5, 21])**.** There is a randomized algorithm $\mathsf{DelTrap}$ $(\mathbf{A}' = (\mathbf{A}|\mathbf{A}_1), \mathbf{R}, H', s')$ that given a vector $\mathbf{A}' = (\mathbf{A}|\mathbf{A}_1) \in \mathcal{R}_q^{1\times m'}$ (where $\mathbf{A}_1 \in \mathcal{R}_q^{1\times \overline{m}}$ and $m' \geqslant m+\overline{m}$), a trapdoor $\mathbf{R} \in \mathcal{R}^{(m-k)\times k}$ corresponding to the vector $\mathbf{A} \in \mathcal{R}_q^{1\times m}$, an invertible element $H' \in \mathcal{R}_q$, and Gaussian parameter $s' \geqslant \eta_\varepsilon(\Lambda^\perp(\mathbf{A}))$. It outputs a trapdoor $\mathbf{R}' \in \mathcal{R}_q^{m\times \overline{m}}$ for the vector $\mathbf{A}'$, under the tag H'.

Lemma 7 (RandBasis [5])**.** There is a randomized algorithm $\mathsf{RandBasis}(\mathbf{A}, \mathbf{R}, \tau)$ that given a vector $\mathbf{A} \in \mathcal{R}_q^{1\times m}$ with its trapdoor $\mathbf{R} \in \mathcal{R}^{(m-k)\times k}$, and a Gaussian parameter $\tau = \|\widetilde{\mathbf{R}}\|\omega(\sqrt{\log mn})$, outputs a trapdoor $\mathbf{R}'$ sampled from a distribution $\mathcal{D}_{\mathcal{R}^{(m-k)\times k},\tau}$.

In particular, $\mathsf{RandBasis}$ algorithm is only used for our security proof.

Appendix D.2 Technique Overview

Modified PAEKS. Since the PAEKS scheme only achieves a one-to-one search model, its application is somewhat limited. We can trivially adapt PAEKS to multi-user settings. Informally, public keys $\mathbf{A}_{r_1}, ..., \mathbf{A}_{r_l}$ of l receivers are selected in the encryption phase. The sender generates a ciphertext component $\mathbf{C}_i = \mathbf{A}_{r_i}^T s + \mathbf{e}_i$ for $i \in [1, l]$ so that these receivers can search for the ciphertext. Nevertheless, it still requires the sender to specify the receivers in advance and lacks flexibility. Therefore, we would like to realize a flexible keyword search without pre-fixing target receivers.

Our inspiration comes from a fully key-homomorphic ABE scheme [4]. We follow this one and adopt an access policy to control the receiver's search permission. Our construction utilizes three deterministic algorithms: $\mathsf{EVAL}_{MPK}, \mathsf{EVAL}_C$, and EVAL_{SIM} introduced by [4], with a family of policies $\mathcal{F} : \{0,1\}^l \to \{0,1\}$ with depth d.

- Master public key evaluation: For $i \in [1, l]$, let $\mathbf{B}_i$ be a random row vector, we have $\mathsf{EVAL}_{MPK}(f \in \mathcal{F}, (\mathbf{B}_1, \cdots, \mathbf{B}_l)) \to \mathbf{B}_f$.
- Ciphertext evaluation: For $i \in [1, l]$, let $\mathbf{x} = (x_1, \cdots, x_l)$ be an attribute vector and $\mathbf{C}_i = (\mathbf{B}_i + (\mathbf{0}|x_i\mathbf{G}))^T s + \mathbf{e}_i$ be a ciphertext component, then we have $\mathsf{EVAL}_C\left(f \in \mathcal{F}, (x_i, \mathbf{B}_i, \mathbf{C}_i)_{i=1}^l\right) \to \mathbf{C}_f$. By [1](Lemma 15), we can know $\|\mathbf{e}_i\| = \|\mathbf{S}_i\mathbf{e}_A\| \leqslant C_1 \cdot t\sigma m\sqrt{2n}$, where $\mathbf{S}_i \xleftarrow{R} \{-1,1\}^{m\times m}$, $\mathbf{e}_A \xleftarrow{R} D_{\mathcal{R}^m,\sigma}$ and C_1 is a universal constant. Thus the evaluated ciphertext $\mathbf{C}_f = (\mathbf{B}_f + (\mathbf{0}|f(\mathbf{x})\mathbf{G}))^T s + \mathbf{e}_f$ and $\|\mathbf{e}_f\| < C_1 \cdot t\sigma m\sqrt{2n}(mn)^{O(d)}$.
- Simulation evaluation: For $i \in [1, l]$, we have $\mathsf{EVAL}_{SIM}(f \in \mathcal{F}, (x_i^*, \mathbf{S}_i)_{i=1}^l, \mathbf{A}) \to \mathbf{S}_f$, where $\mathbf{x}^* = (x_1^*, \cdots, x_l^*)$ is a challenge attribute, $\mathbf{S}_i \xleftarrow{R} \{-1,1\}^{m\times m}$ is a random matrix and $\mathbf{A}$ is a random vector. The algorithm EVAL_{SIM} satisfies $\mathbf{AS}_f - (\mathbf{0}|f(\mathbf{x}^*)\mathbf{G}) = \mathbf{B}_f$, where $\mathbf{B}_f = \mathsf{EVAL}_{MPK}(f \in \mathcal{F}, (\mathbf{AS}_1 - (\mathbf{0}|x_1^*\mathbf{G}), \cdots, \mathbf{AS}_l - (\mathbf{0}|x_l^*\mathbf{G})))$ and $\|\mathbf{S}_f\| \leqslant C_1 \cdot \sqrt{2m}(mn)^{O(d)}$.

Using key-homomorphic techniques, we could embed an access policy f and an attribute vector $\mathbf{x}$ into the receiver's secret key and the ciphertext, respectively. In this way, if and only if $f(\mathbf{x}) = 0$, the receiver can search for ciphertexts. Our construction idea is described as follows: The key generation center (KGC) produces a master key pair and then generates the sender's key pair via the identity label ID and the $\mathsf{TrapDel}$ algorithm. We refer to ID as the sender's public key and to a trapdoor for $(\mathbf{A}|H_3(ID))$ as the sender's secret key, where $\mathbf{A}$ is a (pseudo)random vector with the trapdoor. For the receiver's secret key, KGC outputs a trapdoor for $(\mathbf{A}|\mathbf{B}_f)$ by running $\mathsf{TrapDel}$ algorithm. During the encryption phase, there are three ciphertext components, $\mathbf{C}_1, C_2$ and $\mathbf{C}_3$, with regular structures that follow the constructions of [1, 5]. They hide the information of the keyword. The sender in the system computes extra components $\mathbf{C}_i = (\mathbf{B}_i + (\mathbf{0}|x_i\mathbf{G}))^T s + \mathbf{e}_i$ for an attribute vector $\mathbf{x} = (x_1, ..., x_l)$, where s is a random polynomial, and $\mathbf{e}_i$ is an error vector for $i \in [1, l]$. Besides, let $\mathbf{F}_{ID,b} = (\mathbf{A}|H_3(ID)|H_4(ID, b))$, the sender samples a short vector $\mathbf{v}$ via the $\mathsf{SamplePre}$ algorithm such that $\mathbf{F}_{ID,b}\mathbf{v} = H_2(\mathbf{C}_1, b)$, where b is regarded as a secret polynomial. Therefore, $(\mathbf{C}_1, C_2, \mathbf{C}_3, \{\mathbf{C}_i\}_{i=1}^l, \mathbf{v})$ are overall ciphertexts that the sender outputs. After that, the receiver runs $\mathsf{SamplePre}$ algorithm to evaluate $\boldsymbol{\varphi}_f$ such that $(\mathbf{A}|\mathbf{B}_f)\boldsymbol{\varphi}_f = t$, in which t is a value related to the keyword. Finally, $\{\mathbf{C}_i\}_{i=1}^l$ are homomorphic evaluated over the policy f, and the cloud server outputs a test result by matching the ciphertext and the token.

Appendix D.3 Correctness Analysis

To show the correctness of the AB-AEKS scheme, first note that we know $\mathbf{C}_f$ is homomorphically evaluated over the policy circuit f when $f(\mathbf{x}) = 0$. Let $\boldsymbol{\varphi}_f^T = \left(\boldsymbol{\varphi}_A^T|\boldsymbol{\varphi}_B^T\right)$ and $\mathbf{p}^T = \left(\mathbf{p}_0^T|\mathbf{p}_1^T\right)$, where $\boldsymbol{\varphi}_A, \boldsymbol{\varphi}_B \in \mathcal{R}_q^m$, $\mathbf{p}_0 \in \mathcal{R}_q^{m-k}$ and $\mathbf{p}_1 \in \mathcal{R}_q^k$. Indeed, if the ciphertext and token contain the same keyword, we have

$$\begin{aligned}
&\left(\boldsymbol{\varphi}_f^T|\mathbf{p}^T\right)(\mathbf{C}_3|\mathbf{C}_f|\mathbf{C}_1)\\
&= \boldsymbol{\varphi}_A^T\left(\mathbf{A}^T s + \mathbf{e}_A\right) + \boldsymbol{\varphi}_B^T\left(\mathbf{B}_f^T s + \mathbf{e}_f\right) + \mathbf{p}^T\left(\mathbf{F}_w^T s + \left(\mathbf{e}_0^T|\mathbf{e}_1^T\right)^T\right)\\
&= (\mathbf{A}\boldsymbol{\varphi}_A + \mathbf{B}_f\boldsymbol{\varphi}_B + \mathbf{F}_w\mathbf{p})^T s + \varphi_{\mathbf{A}}^T\mathbf{e}_A + \boldsymbol{\varphi}_B^T\mathbf{e}_f + \left(\mathbf{p}_0^T|\mathbf{p}_1^T\right)\left(\mathbf{e}_0^T|\mathbf{e}_1^T\right)^T\\
&= u \cdot s + \boldsymbol{\varphi}_A^T\mathbf{e}_A + \boldsymbol{\varphi}_B^T\mathbf{e}_f + \mathbf{p}_0^T\mathbf{e}_0 + \mathbf{p}_1^T\mathbf{e}_1.
\end{aligned}$$

If the error term $e - \boldsymbol{\varphi}_A^T\mathbf{e}_A - \boldsymbol{\varphi}_B^T\mathbf{e}_f - \mathbf{p}_0^T\mathbf{e}_0 - \mathbf{p}_1^T\mathbf{e}_1$ has l_2 norm less than $\lfloor \frac{q}{4} \rfloor$, then we capture the polynomial b such that $\mathbf{F}_{ID,b}\mathbf{v} = h$ holds. Furthermore, $\mathbf{v}$ is an integer vector of size $3mn$ with Gaussian parameter ζ, which is sampled from the $\mathsf{SamplePre}$ algorithm, so we have $0 < \|\mathbf{v}\| \leqslant t\zeta\sqrt{3mn}$ with an overwhelming advantage. Thus the Test algorithm yields a correct result.

Appendix D.4 Performance Comparison

We compare the functional and storage aspects of the related schemes [9, 11, 13, 15, 16] with ours, in Table D2.

From the perspective of function, our AB-AEKS scheme is proved to be secure, i.e., IND-CKA and UNF-IKGA, based on the two assumptions of ring variants and satisfies fine-grained searchability and quantum-resistant. Regarding storage, our ciphertexts are larger than those of schemes [9,13] under the classical problems, where [9] does not have the property of fine-grained. However, for the size of the token, our scheme remains the same order of magnitude as other schemes except for [9]. The results show that our scheme has functional and efficiency advantages compared to others.

§) ***Note.*** IND: Indistinguishable; MIND: Multi-ciphertext Indistinguishable; UNF: Unforgeable; TPA: Trapdoor Privacy Attack; CKA: Chosen-Keyword Attacks; IKGA: Insider Keyword Guessing Attacks; CCA: Chosen-Ciphertext Attacks; CMA: Chosen-Message Attacks; N: lattice dimension in [15]; s, k: the length of binary plaintext string and the dimension of the attribute vector in [11], respectively; m: lattice parameters, i.e., random matrices $\mathbf{A}_1, \cdots, \mathbf{A}_k \in \mathbb{Z}_q^{n\times m}$; l_e: the dimension of the matrix $\mathbf{D}$ in [13], where $\mathbf{D}$ denotes its decryption predicate; l: the number of attributes.

Table D2 The functional and storage comparison

Feature \ Scheme	[9]	[15]	[16]	[11]	[13]	Ours
Fine-grained	✗	✗	✗	✓	✓	✓
Quantum-resistant	✗	✓	✓	✓	✗	✓
Security model	IND-TPA, IND-IKGA	IND-CKA, IND-IKGA	MIND-CKA, MIND-IKGA	IND-CKA, UNF-IKGA	IND-CCA, UNF-CMA, IND-CKA	IND-CKA, UNF-IKGA
Hardness problem	DBDH, DLIN	/	/	LWE,SIS	n-DBDHE, n-CDHE,DL	Ring-LWE, Ring-ISIS
Ciphertext size	$2\|\mathbb{G}_1\|$	$2N\|q\| + N\|\mathbb{Z}_q\|$	$\rho(2m+1)\|\mathbb{Z}_q\|$	$(s+(k+2)ms+m)\|\mathbb{Z}_q\|$	$6\|\mathbb{G}_1\|$	$((5+l)m+1)\|\mathcal{R}_q\|$
Token size	$\|\mathbb{G}_T\|$	$N\|q\|$	$2m\|\mathbb{Z}_q\|$	$3m\|\mathbb{Z}_q\|$	$(2l_e+2)\|\mathbb{G}_1\|$	$3m\|\mathcal{R}_q\|$

Appendix E Proof of Theorem 2

In this section, we give the security proof and the parameter choices for our AB-AEKS scheme.

Appendix E.1 Security Proof

Proof Sketch. We propose the security models that are aimed at two kinds of attacks: chosen-keyword attacks (CKA) and insider keyword guessing attacks (IKGA) as follows:

Firstly, we prove that our scheme is IND-CKA security based on the ring-LWE assumption. To achieve this goal, we require that the adversary submits a target attribute vector $\mathbf{x}^* = (x_1^*, ..., x_l^*) \in \{0,1\}^l$ before being given MPK. More precisely, the ring-LWE instance is constructed to a random vector $\overline{\mathbf{A}}$ which is used to simulate the output of $\mathsf{TrapGen}$ algorithm (where the tag is 0). Since the (pseudo)random property of the $\mathsf{TrapGen}$ algorithm, the simulation tuple (where the tag is 0) is indistinguishable from $(\mathbf{A}, \mathbf{R_A})$ (where the tag is 1 in the real scheme). The challenger then samples $l+1$ random matrices $\widehat{\mathbf{S}}$ and $\mathbf{S}_i$, such that $\widehat{\mathbf{A}} = \mathbf{A}\widehat{\mathbf{S}} - (\mathbf{0}|p^*\mathbf{G})$ and $\mathbf{B}_i = \mathbf{A}\mathbf{S}_i - (\mathbf{0}|x_i^*\mathbf{G})$ for $p^* \in \mathbb{Z}_q, i \in [1,l]$. The distribution $(\mathbf{A}, \mathbf{A}\mathbf{S}_i, \mathbf{e}_i)$ is statistically close to the distribution $(\mathbf{A}, \mathbf{A}', \mathbf{e}_i)$, where $\mathbf{A}'$ is a random vector. In other words, the distribution of $\mathbf{A}\widehat{\mathbf{S}}$ and $\mathbf{A}\mathbf{S}_i$ are statistically close to the uniform, thus $\widehat{\mathbf{A}}, \mathbf{B}_i$ and uniformly random vectors are indistinguishable.

Subsequently, the adversary initiates oracle queries of hash $\mathcal{O}_{H_1}$ and $\mathcal{O}_{H_2}$, key generation $\mathcal{O}_{\mathsf{KeyGen}_s}$ and $\mathcal{O}_{\mathsf{KeyGen}_r}$, encryption $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$ and token $\mathcal{O}_{\mathsf{Token}}$. In particular, the challenger could answer the adversary even if $f(\mathbf{x}^*) = 0$ in $\mathcal{O}_{\mathsf{Token}}$ because it sets $\mathbf{F}_w = \mathbf{A}\widehat{\mathbf{S}} + (\mathbf{0}|(H_1(w) - p^*)\mathbf{G})$ and bypasses the receiver's secret key by using a trapdoor for $(\mathbf{A}|\mathbf{F}_w)$. Finally, receiving a target keyword and an identity, the challenger flips a coin $r \in \{0,1\}$ and returns a challenge ciphertext $C^* = \left(\mathbf{C}_1^*, C_2^*, \mathbf{C}_3^*, \{\mathbf{C}_i^*\}_{i=1}^l, \mathbf{v}^*\right)$ to adversary. Note that if $r = 1$, the challenger simulates

$$\mathbf{C}_1^* = \widehat{\mathbf{S}}^T\mathbf{C}_3^*, C_2^* = b_0 + b^*\left\lfloor\frac{q}{2}\right\rfloor, \mathbf{C}_3^* = \left(\overline{\mathbf{B}}^T| - \overline{\mathbf{B}}^T\mathbf{R_A} + \hat{\mathbf{e}}^T\right)^T, \mathbf{C}_i^* = \mathbf{S}_i^T\mathbf{C}_3^*,$$

which are consist of ring-LWE instances. Otherwise, the challenge ciphertext is a random tuple. Besides, since h^* is randomly selected, the adversary is also unable to distinguish $(h^*, \mathbf{v}^*)$ from a uniformly random distribution.

After that, we describe the UNF-IKGA security proof for our scheme. We require that the adversary initiates two lists L_{ID} and L_b that it makes queries, and submits a target attribute vector before giving MPK. Let $(\mathbf{A}, y)$ be a ring-ISIS instance, the challenger programs y into the response for H_4 oracle. It then sets $\mathbf{B}_i$ same as in above security game except that $\widehat{\mathbf{A}} = \mathbf{A}\widehat{\mathbf{S}}$. For each H_2 query on an identity ID, the challenger programs $H_2(ID) = (\mathbf{A}|\mathbf{T}_{ID}) - (\mathbf{0}|\mathbf{G})$ if $ID \in L_{ID}$; otherwise, $H_2(ID) = (\mathbf{A}|\mathbf{T}_{ID})$. A similar programming method is applied for H_3: for every pair $(ID, b) \in L_b$, the hash value is programmed as $H_3(ID, b) = (\mathbf{A}|\mathbf{T}_b) - (\mathbf{0}|\mathbf{G})$, other queries are programmed as $H_3(ID, b) = (\mathbf{A}|\mathbf{T}_b)$. By this way, the challenger could response all queries of the adversary in $\mathcal{O}_{\mathsf{AB-AEKS}}$ and $\mathcal{O}_{\mathsf{Token}}$. In the end the adversary returns a forgery C^* based on $(\mathbf{x}^*, w^*, ID^*)$. We suppose that the adversary queried $H_2(ID^*)$ and $H_3(ID^*, b)$.

Theorem 2 contains the two security requirements of the AB-AEKS scheme, and for clarity, we argue for the scheme's security through the following two lemmas.

Lemma 8. If the hardness of $Ring\text{-}LWE_{n,q,m,D_{\mathcal{R},\tau}}$ problem holds, our proposed AB-AEKS scheme is proved to be selective-attribute IND-CKA security, in the random oracle model.

Proof. The proof is described in a security game between a PPT adversary $\mathcal{A}$ and a challenger $\mathcal{C}$.

Init. $\mathcal{A}$ chooses a target attribute vector $\mathbf{x}^* = (x_1^*, \cdots, x_l^*) \in \{0,1\}^l$.

Setup. $\mathcal{C}$ queries a ring-LWE oracle $m-k+1$ times and receives some samples $(a_i, b_i) \in \mathcal{R}_q \times \mathcal{R}_q$, for $0 \leqslant i \leqslant m-k$. It selects hash functions $H_1 : \{0,1\}^n \to \mathbb{Z}_q$, $H_2 : \{0,1\}^* \to \mathcal{R}_q^{1\times m}$, $H_3 : \{0,1\}^* \times \mathcal{R}_2 \to \mathcal{R}_q^{1\times m}$, $H_4 : \mathcal{R}_q^m \times \mathcal{R}_2 \to \mathcal{R}_q$, and sets a list L_1. Denoted by q_1 is the maximum number of queries to H_1, $\mathcal{C}$ chooses an integer $j^* \in [1, q_1]$. Let $\overline{\mathbf{A}} = (a_1, \cdots, a_{m-k})^T$, $\overline{\mathbf{B}} = (b_1, \cdots, b_{m-k})^T$ from $m-k$ of given ring-LWE samples, it sets $u = a_0$. Running $(\mathbf{A}, \mathbf{R_A}) \leftarrow \mathsf{TrapGen}(\overline{\mathbf{A}}, H = 0, \sigma, q)$, and sampling $l+1$ random matrices $\widehat{\mathbf{S}}, \mathbf{S}_i \xleftarrow{R} \{-1,1\}^{m\times m}$ and a random element $p^* \xleftarrow{R} \mathbb{Z}_q$, it sets $\widehat{\mathbf{A}} = \mathbf{A}\widehat{\mathbf{S}} - (\mathbf{0}|p^*\mathbf{G})$ and $\mathbf{B}_i = \mathbf{A}\mathbf{S}_i - (\mathbf{0}|x_i^*\mathbf{G})$ for $i \in [1,l]$. $\mathcal{C}$ outputs $MPK = \left(\mathbf{A}, \widehat{\mathbf{A}}, \{\mathbf{B}_i\}_{i=1}^l, u, H_1, H_2, H_3, H_4\right)$ and send MPK to adversary.

Query phase 1. $\mathcal{A}$ issues the following queries:

- $\mathcal{O}_{H_1}$: For the j-th query on w, where $j \in [1, q_1]$. If $j = j^*$ such that $w = w^*$, $\mathcal{C}$ returns $H_1(w) = p^*$ to $\mathcal{A}$ and adds (w^*, p^*) to the list L_1. Otherwise, $\mathcal{C}$ checks whether the hash value of w has been queried before in list L_1. If yes, $\mathcal{C}$ returns the previous value; if not, it selects randomly $p \in \mathbb{Z}_q$, returns p, and adds (w, p) to the list L_1.
- $\mathcal{O}_{H_2}$: For each identity ID, $\mathcal{C}$ samples a short $\mathbf{T}_{ID} \xleftarrow{R} \mathcal{D}_{\mathcal{R}^{m\times m}, s'}$, returns $H_2(ID) = \mathbf{A}\mathbf{T}_{ID} - (\mathbf{0}|\mathbf{G})$, and stores $(ID, \mathbf{T}_{ID})$.
- $\mathcal{O}_{\mathsf{KeyGen}_s}$: $\mathcal{A}$ could query the sender's key pair under an identity ID. $\mathcal{C}$ first checks ID was previously queried in $\mathcal{O}_{H_2}$. If yes, it sets $(PK_s, SK_s) = (ID, \mathbf{T}_{ID})$. Otherwise, $\mathcal{C}$ samples a short $\mathbf{T}_{ID} \xleftarrow{R} \mathcal{D}_{\mathcal{R}^{m\times m}, s'}$ and sets $H_2(ID) = \mathbf{A}\mathbf{T}_{ID} - (\mathbf{0}|\mathbf{G})$. Hence, $\mathbf{T}_{ID}$ is a trapdoor for $\mathbf{F}_{ID} = (\mathbf{A}|H_2(ID))$ and $\mathcal{C}$ returns $(PK_s, SK_s) = (ID, \mathbf{T}_{ID})$ to $\mathcal{A}$.
- $\mathcal{O}_{\mathsf{KeyGen}_r}$: $\mathcal{A}$ could query the receiver's secret key under an access policy f, except for $f(\mathbf{x}^*) = 0$. $\mathcal{C}$ computes

$$\mathbf{S}_f \leftarrow \mathsf{EVAL}_{SIM}(f, (x_i^*, \mathbf{S}_i)_{i=1}^l, \mathbf{A})$$

and

$$\mathbf{B}_f \leftarrow \mathsf{EVAL}_{MPK}(f, (\mathbf{AS}_i - (\mathbf{0}|x_i^*\mathbf{G}))_{i=1}^l),$$

which satisfies $\mathbf{AS}_f - (\mathbf{0}|\mathbf{G}) = \mathbf{B}_f$. Hence, $\mathbf{S}_f$ is a trapdoor for $(\mathbf{A}|\mathbf{B}_f)$ and $\mathcal{C}$ returns $\mathbf{T}_f \leftarrow \mathsf{RandBasis}(\mathbf{A}|\mathbf{B}_f, \mathbf{S}_f, \tau)$ to $\mathcal{A}$.

- $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$: $\mathcal{A}$ could make the encryption query on a keyword w, an attribute vector $\mathbf{x}$ and identity ID. $\mathcal{C}$ computes the ciphertext C in a normal way. The specific steps are described below.
  1) If $w \neq w^* \wedge \mathbf{x} = \mathbf{x}^*$, we have $\mathbf{F}_w = \mathbf{A}\widehat{\mathbf{S}} + (\mathbf{0}|(H_1(w) - p^*)\mathbf{G})$ and $\{\mathbf{C}_i = (\mathbf{AS}_i)^T s + \mathbf{e}_i\}_{i=1}^l$.
  2) If $\mathbf{x} \neq \mathbf{x}^* \wedge w = w^*$ (i.e., at least exists i s.t. $x_i \neq x_i^*$), we have $\mathbf{F}_w = \mathbf{A}\widehat{\mathbf{S}}$ and $\{\mathbf{C}_i = (\mathbf{AS}_i + (\mathbf{0}|(x_i - x_i^*)\mathbf{G}))^T s + \mathbf{e}_i\}_{i=1}^l$.
  3) If $w = w^* \wedge \mathbf{x} = \mathbf{x}^*$, we have $\mathbf{F}_w = \mathbf{A}\widehat{\mathbf{S}}$ and $\{\mathbf{C}_i = (\mathbf{AS}_i)^T s + \mathbf{e}_i\}_{i=1}^l$.

  Furthermore, $\mathcal{C}$ captures $(PK_s, SK_s) = (ID, \mathbf{T}_{ID})$ same as $\mathcal{O}_{\mathsf{KeyGen}_s}$, computes $\mathbf{F}_{ID,b} = (\mathbf{A}|H_2(ID)|H_3(ID,b))$ and $h = H_4(\mathbf{C}_1, b)$, where $b \in \mathcal{R}_2$. It samples $\mathbf{v}$ by running $\mathsf{SamplePre}$ algorithm such that $\mathbf{F}_{ID,b}\mathbf{v} = h$, and returns the ciphertext C to $\mathcal{A}$.
- $\mathcal{O}_{\mathsf{Token}}$: $\mathcal{A}$ could query the token of a keyword w related to an access policy f, with the restriction that $f(\mathbf{x}^*) \neq 0$ or $f(\mathbf{x}^*) = 0 \wedge w \neq w^*$.
  1) If $f(\mathbf{x}^*) \neq 0$, there exists $\mathbf{T}_f$ corresponding to the policy f. $\mathcal{C}$ computes the token K in a normal way.
  2) Else, $f(\mathbf{x}^*) = 0$ but $w \neq w^*$. Since $f(\mathbf{x}^*) = 0$, there is $\mathbf{AS}_f = \mathbf{B}_f$. It means that $\mathcal{C}$ does not generate a trapdoor for $(\mathbf{A}|\mathbf{B}_f)$. However, we have $w \neq w^*$, it shows $\mathbf{F}_w = \mathbf{A}\widehat{\mathbf{S}} + (\mathbf{0}|(H_1(w) - p^*)\mathbf{G})$. Hence $\widehat{\mathbf{S}}$ is a trapdoor for $(\mathbf{A}|\mathbf{F}_w)$. $\mathcal{C}$ computes $\mathbf{T}_{\mathbf{A}|\mathbf{F}_w|\mathbf{B}_f} \leftarrow \mathsf{DelTrap}(\mathbf{A}|\mathbf{F}_w|\mathbf{B}_f, \mathbf{T}_{\mathbf{A}|\mathbf{F}_w}, H' = 1, s')$, where $\mathbf{T}_{\mathbf{A}|\mathbf{F}_w} \leftarrow \mathsf{RandBasis}(\mathbf{A}|\mathbf{F}_w, \widehat{\mathbf{S}}, \tau)$. Based on this trapdoor $\mathbf{T}_{\mathbf{A}|\mathbf{F}_w|\mathbf{B}_f}$, $\mathcal{C}$ captures a vector $\boldsymbol{\varphi}$ via the $\mathsf{SamplePre}$ algorithm such that $(\mathbf{A}|\mathbf{F}_w|\mathbf{B}_f)\boldsymbol{\varphi} = u$. Let $\boldsymbol{\varphi} = (\boldsymbol{\varphi}_1; \boldsymbol{\varphi}_2; \boldsymbol{\varphi}_3)$, we permute the resulting vector to reach $(\mathbf{A}|\mathbf{B}_f|\mathbf{F}_w)\boldsymbol{\varphi}' = u$, where $\boldsymbol{\varphi}' = (\boldsymbol{\varphi}_1; \boldsymbol{\varphi}_3; \boldsymbol{\varphi}_2)$. So we can see that the token K is given by $\boldsymbol{\varphi}_f^T = (\boldsymbol{\varphi}_1^T|\boldsymbol{\varphi}_3^T)$ and $\mathbf{p}^T = \boldsymbol{\varphi}_2^T$.

Challenge. $\mathcal{A}$ transmits a keyword w^* and an identity ID^*. $\mathcal{C}$ chooses randomly a bit $r \in \{0,1\}$ and returns a random ciphertext C^* if $r = 0$. Otherwise, $\mathcal{C}$ selects randomly a polynomial $b^* \in \mathcal{R}_2$, computes $C_2^* = b_0 + b^* \lfloor \frac{q}{2} \rfloor$ and $\mathbf{C}_3^* = \left(\overline{\mathbf{B}}^T| - \overline{\mathbf{B}}^T\mathbf{R}_\mathbf{A} + \hat{\mathbf{e}}^T\right)^T$, where $\hat{\mathbf{e}} \xleftarrow{R} D_{\mathcal{R}^k,\mu}$ for some μ real. Setting $\mathbf{C}_i^* = \mathbf{S}_i^T\mathbf{C}_3^*$ for $i \in [1,l]$, and $\mathbf{C}_1^* = \widehat{\mathbf{S}}^T\mathbf{C}_3^*$. Let $\mathbf{F}_{ID^*,b^*} = (\mathbf{A}|H_2(ID^*)|H_3(ID^*,b^*))$ and $h^* = H_4(\mathbf{C}_1^*, b^*)$, it evaluates $\mathbf{v}^*$ from $\mathsf{TrapDel}$ and $\mathsf{SamplePre}$ algorithms by using the trapdoor of $(\mathbf{A}|H_2(ID^*))$. Then, it returns $C^* = \left(\mathbf{C}_1^*, C_2^*, \mathbf{C}_3^*, \{\mathbf{C}_i^*\}_{i=1}^l, \mathbf{v}^*\right)$ to adversary.

Query phase 2. $\mathcal{A}$ continues to query oracle as phase 1.

Guess. $\mathcal{A}$ outputs a guess $r' \in \{0,1\}$, it wins the security game if $r' = r$ with overwhelming probability.

For $i \in [0, m-k]$, if samples (a_i, b_i) are drawn from the ring-LWE distribution, we have $\overline{\mathbf{B}} = \overline{\mathbf{A}}s + \overline{\mathbf{e}}$ and $b_0 = a_0 s + e_0$ for $s \in \mathcal{R}_q$, $\overline{\mathbf{e}} \xleftarrow{R} D_{\mathcal{R}^{m-k},\tau}$, $e_0 \xleftarrow{R} D_{\mathcal{R},\tau}$. The challenge ciphertext is replaced as

$$C_2^* = b_0 + b^* \left\lfloor \frac{q}{2} \right\rfloor = u \cdot s + e_0 + b^* \left\lfloor \frac{q}{2} \right\rfloor$$

and

$$\mathbf{C}_3^* = \left(\overline{\mathbf{B}}^T| - \overline{\mathbf{B}}^T\mathbf{R}_\mathbf{A} + \hat{\mathbf{e}}^T\right)^T = \mathbf{A}^T s + \left(\overline{\mathbf{e}}^T| - \overline{\mathbf{e}}^T\mathbf{R}_\mathbf{A} + \hat{\mathbf{e}}^T\right)^T.$$

For $i \in [1,l]$,

$$\mathbf{C}_i^* = \mathbf{S}_i^T\mathbf{C}_3^* = (\mathbf{B}_i + (\mathbf{0}|x_i^*\mathbf{G}))^T s + \mathbf{S}_i^T\left(\overline{\mathbf{e}}^T| - \overline{\mathbf{e}}^T\mathbf{R}_\mathbf{A} + \hat{\mathbf{e}}^T\right)^T$$

and

$$\mathbf{C}_1^* = \widehat{\mathbf{S}}^T\mathbf{C}_3^* = (\widehat{\mathbf{A}} + (\mathbf{0}|p^*\mathbf{G}))^T s + \widehat{\mathbf{S}}^T\left(\overline{\mathbf{e}}^T| - \overline{\mathbf{e}}^T\mathbf{R}_\mathbf{A} + \hat{\mathbf{e}}^T\right)^T.$$

Now we consider the error term. If fixed $\overline{\mathbf{e}}$, the distribution of $-\overline{\mathbf{e}}^T\mathbf{R}_\mathbf{A} + \hat{\mathbf{e}}^T$ is indistinguishable from $\mathcal{D}_{\mathcal{R}^k,\gamma}$, where $\gamma^2 = (\sigma\|\overline{\mathbf{e}}\|)^2 + \mu^2$. The challenge ciphertext C^* still remains the same format as real scheme. If samples (a_i, b_i) are chosen from a uniformly random distribution, then C^* is indistinguishable from the uniform distribution. ■

Remark 1. *Supposing the adversary breaks our scheme under IND-CKA with a non-negligible probability ε, and w^* is indeed the j^*-th query in H_1 queries with probability $1/q_1$. The challenger has advantage at least $\varepsilon' = \varepsilon/2q_1$ in solving the ring-LWE problem.*

Lemma 9. If the hardness of *Ring-ISIS*$_{q,m,\beta}$ problem holds, our proposed AB-AEKS scheme is proved to be selective-attribute UNF-IKGA, in the random oracle model.

Proof. The security proof is described a game between the PPT adversary $\mathcal{A}$ and the challenger $\mathcal{C}$.

Init. Denote the list of all identities ID for sender's key queries as L_{ID}, and the list of all identity-polynomial pairs (ID, b) for encryption queries as L_b. $\mathcal{A}$ outputs these two lists and an attribute vector $\mathbf{x}^* = (x_1^*, ..., x_l^*) \in \{0,1\}^l$.

Setup. $\mathcal{C}$ queries a ring-ISIS oracle $m+1$ times and receives the samples $(\mathbf{U}, y) = (u_1, u_2, \cdots, u_m, y) \in \mathcal{R}_q^{1\times m} \times \mathcal{R}_q$, and it attempts to find a polynomial $\mathbf{z} \in \mathcal{R}^m$ such that $\mathbf{Uz} = y$ and $0 < \|\mathbf{z}\| \leqslant \eta$. Denoted by L_4 is a list and q_4 is the maximum number of queries to H_4 that the adversary makes, $\mathcal{C}$ chooses an integer $j^* \in [1, q_4]$. Let $\mathbf{A} = \mathbf{U}$, similar to Lemma 8, it samples $l+1$ random matrices $\widehat{\mathbf{S}}, \mathbf{S}_i \xleftarrow{R} \{-1,1\}^{m\times m}$, sets $\widehat{\mathbf{A}} = \mathbf{A}\widehat{\mathbf{S}}$ and $\mathbf{B}_i = \mathbf{AS}_i - (\mathbf{0}|x_i^*\mathbf{G})$ for $i \in [1,l]$. $\mathcal{C}$ outputs $MPK = \left(\mathbf{A}, \widehat{\mathbf{A}}, \{\mathbf{B}_i\}_{i=1}^l, u, H_1, H_2, H_3, H_4\right)$ to adversary.

Query phase. $\mathcal{A}$ issues the following queries:

- $\mathcal{O}_{H_2}$: For each identity $ID \in L_{ID}$, $\mathcal{C}$ samples a short $\mathbf{T}_{ID} \xleftarrow{R} \mathcal{D}_{\mathcal{R}^{m\times m},s'}$ and returns $H_2(ID) = \mathbf{AT}_{ID} - (\mathbf{0}|\mathbf{G})$. Otherwise, the hash value will be programmed to $H_2(ID) = \mathbf{AT}_{ID}$.
- $\mathcal{O}_{H_3}$: For each identity-polynomial pair $(ID, b) \in L_b$, $\mathcal{C}$ samples a short $\mathbf{T}_b \xleftarrow{R} \mathcal{D}_{\mathcal{R}^{m\times m},s'}$ and returns $H_3(ID, b) = \mathbf{AT}_b - (\mathbf{0}|\mathbf{G})$. Otherwise, the hash value will be programmed to $H_3(\mathbf{C}_1, b) = \mathbf{AT}_b$.
- $\mathcal{O}_{H_4}$: For the j-th query on $(\mathbf{C}_1, b)$, where $j \in [1, q_4]$. $\mathcal{C}$ first checks whether the hash value was previously defined in list L_4. If yes, it returns previous value. Otherwise, $\mathcal{C}$ selects randomly $\beta \in \mathcal{R}_q$, adds $(\mathbf{C}_1, b, \beta)$ to list L_4, and returns β. Note that if $j = j^*$, such that $b = b^*$ and $\mathbf{C}_1 = \mathbf{C}_1^*$ which is a required component of the forgery. $\mathcal{C}$ adds $(\mathbf{C}_1^*, b^*, y)$ to the list L_4 and returns y to adversary.

- $\mathcal{O}_{\mathsf{KeyGen}_s}$: Identical to Lemma 8.
- $\mathcal{O}_{\mathsf{KeyGen}_r}$: Identical to Lemma 8.
- $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$: $\mathcal{A}$ could issue the encryption query on a keyword w, an attribute vector $\mathbf{x}$ and identity ID. $\mathcal{C}$ chooses random ring elements and error terms to generate the ciphertext components $(\mathbf{C}_1, C_2, \mathbf{C}_3, \{\mathbf{C}_i\}_{i=1}^{l})$ and h in a normal way. Using $\mathbf{T}_b$ as a trapdoor for $(\mathbf{A}|H_3(ID,b))$, $\mathcal{C}$ computes a trapdoor for $(\mathbf{A}|H_3(ID,b)|H_2(ID))$ via the $\mathsf{TrapDel}$ algorithm. It samples a vector $\mathbf{v}'$ by $\mathsf{SamplePre}$ algorithm such that $(\mathbf{A}|H_3(ID,b)|H_2(ID))\mathbf{v}' = h$, and then permutes $\mathbf{v}'$ to get a vector $\mathbf{v}$. $\mathcal{C}$ returns the ciphertext $C = (\mathbf{C}_1, C_2, \mathbf{C}_3, \{\mathbf{C}_i\}_{i=1}^{l}, \mathbf{v})$ to adversary.
- $\mathcal{O}_{\mathsf{Token}}$: $\mathcal{A}$ could query the token of a keyword w associated with an access policy f. If $f(\mathbf{x}^*) \neq 0$, $\mathcal{C}$ uses the trapdoor $\mathbf{T}_f$ to compute the token K. Otherwise, $f(\mathbf{x}^*) = 0$ but $w \neq w^*$, $\mathcal{C}$ sets $\mathbf{F}_w = \mathbf{A}\widehat{\mathbf{S}} + (\mathbf{0}|H_1(w)\mathbf{G})$ and use the same method as Lemma 8 to get vectors $\boldsymbol{\varphi}_f$ and $\mathbf{p}$. Then it outputs the token K to adversary.

Forgery. $\mathcal{A}$ outputs a forgery $C^* = (\mathbf{C}_1^*, C_2^*, \mathbf{C}_3^*, \{\mathbf{C}_i^*\}_{i=1}^{l}, \mathbf{v}^*)$ associated with $(w^*, \mathbf{x}^*, ID^*)$, with the restriction that $ID^* \notin L_{ID} \wedge (ID^*, b^*) \notin L_b$ and ID^* cannot be queried in $\mathcal{O}_{\mathsf{KeyGen}_s}$ and $\mathcal{O}_{\mathsf{AB\text{-}AEKS}}$. $\mathcal{A}$ wins the game if the forged ciphertext passes Test algorithm.

For the forgery C^*, $\mathcal{C}$ recovers d^* with the purpose of determining b^*, where $d^* = C_2^* - \left(\boldsymbol{\varphi}_f^T|\mathbf{p}^T\right)\left(\mathbf{C}_3^*|\mathbf{C}_f^*|\mathbf{C}_1^*\right)$.

If $\mathcal{A}$ wins the game, which implies that

$$\mathbf{F}_{ID^*,b^*}\mathbf{v}^* = (\mathbf{A}|\mathbf{A}\mathbf{T}_{ID}|\mathbf{A}\mathbf{T}_b)\mathbf{v}^* = \mathbf{U}(\mathbf{I}_m|\mathbf{T}_{ID}|\mathbf{T}_b)\mathbf{v}^* = y.$$

Let $\mathbf{z} = (\mathbf{I}_m|\mathbf{T}_{ID}|\mathbf{T}_b)\mathbf{v}^*$ as its answer to ring-ISIS instance $(\mathbf{U}, y)$ and $\|\mathbf{z}\| = \|(\mathbf{I}_m|\mathbf{T}_{ID}|\mathbf{T}_b)\mathbf{v}^*\| \leqslant (1 + 2ts'm\sqrt{n}) \cdot t\zeta\sqrt{3mn} \leqslant \eta$.■

Remark 2. *The challenger could successfully guess that $H_4(\mathbf{C}_1^*, b^*) = y$ with probability $1/q_4$. If the adversary forges a valid ciphertext with a non-negligible probability ε, the challenger has advantage at least $\varepsilon' = \varepsilon/q_4$ in solving the ring-ISIS problem.*

Appendix E.2 Parameter Choices

In our construction, the parameters of scheme are chosen described as follows:

- n is a power of 2, $m = \mathsf{poly}(n)$, $q \equiv 1 \mod 2n$, $k = \lceil \log_2 q \rceil$, $m = k + 2$.
- The Gaussian parameter for trapdoor generation is $\sigma \approx \sqrt{\ln(2n'/\epsilon)/\pi}$, where n' and ϵ are the maximum dimension of the ring polynomials and the bound on statistical error introduced by each randomized-rounding operation [21], respectively.
- The Gaussian parameter for preimage sampling [3] is $\alpha = \sqrt{5}\sigma$.
- The parameter [3] ζ satisfies $\zeta > s_1(\mathbf{R})\alpha > \sqrt{5}C'\sigma^2(\sqrt{2n} + \sqrt{kn} + t')$, where the universal constant $C' > 0$ (empirically, $C' \approx 1/\sqrt{2\pi}$) and $t' \geqslant 0$.
- By correctness, we require

$$\begin{aligned} &\|e - \boldsymbol{\varphi}_A^T\mathbf{e}_A - \boldsymbol{\varphi}_B^T\mathbf{e}_f - \mathbf{p}_0^T\mathbf{e}_0 - \mathbf{p}_1^T\mathbf{e}_1\| \\ &\leqslant \Delta_\tau + \Delta_\zeta(m\Delta_\sigma + \sqrt{m}\Delta_f) + \Delta_\sigma(2\Delta_\tau + k\Delta_\gamma) < \left\lfloor \frac{q}{4} \right\rfloor, \end{aligned}$$

  where $\Delta_\tau = t\tau\sqrt{n}, \Delta_\zeta = t\zeta\sqrt{n}, \Delta_\sigma = t\sigma\sqrt{n}, \Delta_f = C_1 t\sigma m\sqrt{2n}(mn)^{O(d)}$, and $\Delta_\gamma = t\gamma\sqrt{n}$. For our parameter analysis, [6] shows that taking $C_1 = 12$ is sufficient. Therefore, we choose $q > 5(\Delta_\tau + \Delta_\zeta(m\Delta_\sigma + \sqrt{m}\Delta_f) + \Delta_\sigma(2\Delta_\tau + k\Delta_\gamma))$.
- By security proof, the parameter γ needs to satisfy $\gamma^2 = (\sigma\|\overline{\mathbf{e}}\|)^2 + \mu^2 \leqslant \sigma^2(t\tau\sqrt{2n})^2 + \mu^2$, we choose $\mu = t\sigma\tau\sqrt{2n}$, and then $\gamma = 2\sigma t\tau\sqrt{n}$.
- The Gaussian parameter s' for trapdoor delegation [7, 21] satisfies $s_1(\mathbf{R}) \leqslant s' \cdot O(\sqrt{m} + \sqrt{m})$.

References

1 Agrawal S, Boneh D, Boyen X. Efficient lattice (H)IBE in the standard model. In: Henri G, eds. Advances in Cryptology - EUROCRYPT 2010 - 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Monaco / French Riviera, 2010. 553-572

2 Giuseppe A, Danilo F, David N, et al. Match me if you can: Matchmaking encryption and its applications. In: Alexandra B, Daniele M, eds. Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, 2019. 701-731

3 Pauline B, Pierre A F, Adeline R L, et al. Practical Implementation of Ring-SIS/LWE Based Signature and IBE. In: Tanja L, Rainer S, eds. Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018, Fort Lauderdale, FL, USA, 2018. 271-291

4 Dan B, Craig G, Sergey G, et al. Fully Key-Homomorphic Encryption, Arithmetic Circuit ABE and Compact Garbled Circuits. In: Phong Q N, Elisabeth O, eds. Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, 2014. 533-556

5 David C, Dennis H, Eike K, et al. Bonsai Trees, or How to Delegate a Lattice Basis. In: Henri G, eds. Advances in Cryptology - EUROCRYPT 2010 - 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco / French Riviera, 2010. 523–552

6 Dai W, Yark?n D, Yuriy P, et al. Implementation and Evaluation of a Lattice-Based Key-Policy ABE Scheme. IEEE Trans. Inf. Forensics Secur., 2018, 13: 1169-1184

7 Craig G, Chris P, Vinod V. Trapdoors for hard lattices and new cryptographic constructions. In: Cynthia D, eds. Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, 2008. 197-206

8 He D B, Ma M M, Sherali Z. Certificateless Public Key Authenticated Encryption With Keyword Search for Industrial Internet of Things. IEEE Trans. Ind. Informatics, 2018, 14: 3618-3627

9 Huang Q, Li H B. An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks. Inf. Sci., 2017, 403: 1-14

10 Li H B, Huang Q, Shen J, et al. Designated-server identity-based authenticated encryption with keyword search for encrypted emails. Inf. Sci., 2019, 481: 330-343

11 Li J, Ma M M, Zhang J, et al. Attribute-Based Keyword Search from Lattices. In: Liu Z, Moti Y, eds. Information Security and Cryptology - 15th International Conference, Inscrypt 2019, Nanjing, China, 2019. 66-85

12 Liu X Q, Li H B, Yang G M, et al. Towards Enhanced Security for Certificateless Public-Key Authenticated Encryption with Keyword Search. In: Ron S, Tsz H Y, eds. Provable Security - 13th International Conference, ProvSec 2019, Cairns, QLD, Australia, 2019. 113-129

13 Liu Z H, Fan Y Q. Provably Secure Searchable Attribute-Based Authenticated Encryption Scheme. Int. J. Netw. Secur., 2019, 21: 177-190

14 Liu Z H, Liu Y, Xu J, et al. Verifiable Attribute-based Keyword Search Encryption with Attribute Revocation for Electronic Health Record System. Int. J. Netw. Secur., 2020, 22: 845-856

15 Liu Z Y, Tseng, Y F, Raylin T, et al. Quantum-resistant Public-key Authenticated Encryption with Keyword Search for Industrial Internet of Things. IACR Cryptol. ePrint Arch., 2020, 955

16 Liu Z Y, Tseng, Y F, Raylin T, et al. Public-key Authenticated Encryption with Keyword Search: Cryptanalysis, Enhanced Security, and Quantum-resistant Instantiation. In: Yuji S, Kouichi S, Ding X H, et al., eds. ASIA CCS'22: ACM Asia Conference on Computer and Communications Security, Nagasaki, Japan, 2022. 423-436

17 Vadim L, Daniele M. Generalized Compact Knapsacks Are Collision Resistant. In: Michele B, Bart P, Vladimiro S, et al., eds. Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, 2006. 144-155

18 Vadim L, Chris P, Oded R. On Ideal Lattices and Learning with Errors over Rings. In: Henri G, eds. Advances in Cryptology - EUROCRYPT 2010 - 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco / French Riviera, 2010. 1-23

19 Meng F, Cheng L X, Wang M Q. ABDKS: attribute-based encryption with dynamic keyword search in fog computing. Frontiers Comput. Sci., 2021, 15: 155810

20 Meng F, Cheng L X, Wang M Q. Ciphertext-policy attribute-based encryption with hidden sensitive policy from keyword search techniques in smart city. EURASIP J. Wirel. Commun. Netw., 2021, 2021: 20

21 Daniele M, Chris P. Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller. In: David P, Thomas J, eds. Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, 2012. 700-718

22 Chris P,Alon R. Efficient Collision-Resistant Hashing from Worst-Case Assumptions on Cyclic Lattices. In: Shai H, Tal R, eds. Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, 2006. 145-166

23 Wang H J, Dong X L, Cao Z F. Multi-Value-Independent Ciphertext-Policy Attribute Based Encryption with Fast Keyword Search. IEEE Trans. Serv. Comput., 2020, 13: 1142–1151

24 Yu Y, Shi J B, Li H L, et al. Key-Policy Attribute-Based Encryption With Keyword Search in Virtualized Environments. IEEE J. Sel. Areas Commun., 2020, 38: 1242–1251