Key Encapsulation Mechanism in Ciphertext-policy Attribute based Setting Featuring Revocation and Key-homomorphic Property

Anushree Belel, Ratna Dutta and Sourav Mukhopadhyay
Indian Institute of Technology Kharagpur, Kharagpur, 721302, India

Keywords: Attribute-based Key Encapsulation, Revocation, Key-homomorphism.

Abstract: Cloud computing is a paradigm shift from traditional computing to process, store and share data in an untrusted environment with emerging applications in medical fields, online data storage, social network, big data analysis and online learning platforms. As more and more organizations, business platforms, individuals are choosing cloud, it is very urgent to ensure data security and privacy in the cloud. To safeguard data breaches, it is important to provide fine-grained access control on encrypted data in the cloud. *Ciphertext-policy attribute based encryption* (CP-ABE) is a promising advanced cryptographic primitive that monitors fine-grained access control of sensitive data in untrusted cloud environment. The *revocable* CP-ABE (RCP-ABE) is an extension of CP-ABE which facilitates direct user revocation from the system. In this work, we introduce a refined encapsulated version of RCP-ABE, called *key-homomorphic revocable ciphertext-policy attribute based key encapsulation mechanism* (RCP-ABKEM). Interesting features of this primitive is that it supports *extended correctness* and *key-homomorphism* along with normal *correctness* requirement. Our work is inspired by the work of Sun et al. (PKC 2020) who introduced the notion of *key-homomorphic identity based revocable key encapsulation mechanism* (IRKEM). We generalize the notion of *key-homomorphic* IRKEM in attribute based setting and provide an instantiation of *key-homomorphic* RCP-ABKEM. We support the conjectured security of our candidate by analysis and prove that the scheme achieves selective security against *chosen plaintext attack* (CPA) under the *q-decisional bilinear Diffie-Hellman exponent* (*q*-DBDHE) assumption in the standard model. More interestingly, when contrasted with existing similar scheme, our scheme exhibits better performance over the existing similar schemes in terms of communication overhead and master secret key size and is the first scheme in attribute setting that preserves key homomorphic property. As a refined primitive, *key-homomorphic* RCP-ABKEM is of independent interest and may be utilized as a building block for generic construction of new cryptographic primitive.

1 INTRODUCTION

Identity based encryption (IBE) is an appealing alternative to *public key encryption* (PKE) that removes the requirement of *public key infrastructure* (PKI) by using user's identity like email or IP address as public key. Since the first proposal of IBE by Boneh and Franklin (Boneh and Franklin, 2001), several new advanced cryptographic encryption techniques have been studied extensively by research community including *hierarchical* IBE, *attribute based encryption* (ABE), *predicate encryption* (PE), *functional encryption* (FE), *inner product encryption* (IPE) and so on. ABE is a promising cryptographic primitive to fix the issue of fine-grained access control on encrypted data in one-to-many communications. *Ciphertext policy* ABE (CP-ABE) has huge applications in the context of cloud security, health-record access control, network privacy, data security on mobile devices, Internet of Things (IoT) and many more.

Designing efficient and practical revocation techniques in ABE has been considered to be very important because no organization wants its revoked users to be able to decrypt the data. In general, there are two mechanisms to revoke users in ABE – indirect revocation (Boldyreva et al., 2008) ; (Sahai et al., 2012) and direct revocation (Attrapadung and Imai, 2009); (Liu and Wong, 2015). In indirect revocation approach, an authority periodically updates the secret key only for non-revoked users to enable them to decrypt the encrypted data whereas the revoked users are not allowed to update their secret key. As a consequence, the revoked users fail to decrypt newly generated ciphertext. However, one shortcoming of this approach

Belel, A., Dutta, R. and Mukhopadhyay, S.
Key Encapsulation Mechanism in Ciphertext-policy Attribute based Setting Featuring Revocation and Key-homomorphic Property.
DOI: 10.5220/0011271600003283
In *Proceedings of the 19th International Conference on Security and Cryptography (SECRYPT 2022)*, pages 349-356
ISBN: 978-989-758-590-6; ISSN: 2184-7711

is that it cannot realize instant user revocation. On the contrary, direct revocation lets a public revocation list to be specified during encryption so that the ciphertext can be decrypted only by the users not in the revocation list and whose attributes satisfy the access policy – thus facilitating instant user revocation. Balu and Kuppusamy (Balu and Kuppusamy, 2013) proposed a revocable CP-ABE using direct revocation technique from the hardness of the *decisional bilinear Diffie-Hellman* (DBDH) problem. Wang et al. (Wang et al., 2017) provided a construction of revocable CP-ABE using identity-based revocation from the *modified decisional q-parallel bilinear Diffie-Hellman exponent* (M-*q*-parallel BDHE) assumption. Liu et al. (Liu et al., 2018) proposed a revocable CP-ABE based on direct revocation approach from the *q-decisional bilinear Diffie-Hellman exponent* (*q*-DBDHE) assumption employing a secret key time validation technique to resolve the issue of growth of the revocation list. Liu et al. (Liu et al., 2020) constructed a revocable CP-ABE with ciphertext update from M-*q*-parallel BDHE problem.

Very recently in 2020, Sun et al. (Sun et al., 2020) proposed an encapsulated version of revocable *identity based broadcast encryption* (IBBE) called *key-homomorphic identity based revocable key encapsulation mechanism* (IRKEM) that satisfies *extended correctness* and *key-homomorphism* apart from the normal *correctness* requirement. Their revocation mechanism can be seen as direct revocation in the sense that their approach enables the sender to perform revocation by providing a list of revoked users directly in the ciphertext. This approach does not require any key update procedure and therefore very convenient for regular use. Precisely, they came up with four modular and compact constructions of *key-homomorphic* IRKEM based on the *q-decisional bilinear Diffie-Hellman exponent* (*q*-DBDHE) problem, the *decisional bilinear Diffie-Hellman* (DBDH) problem, the *q-decisional multi-exponent bilinear Diffie-Hellman* (*q*-MEBDH) problem and the *decisional linear* (DLIN) problem in the standard security model. Merging *key-homomorphic* IRKEM with the idea of distributed key-distribution, they also provided a generic construction of *puncturable key encapsulation mechanism* (PKEM) to achieve fine-grained revocation of decryption capability. To support unbounded punctures, their main idea is to generate a share of the encapsulated key on-the-fly and to recover this key from all shares for successful decryption. *Key-homomorphism* property is required for distributing the encapsulated key and *extended correctness* is crucial to compute shares of the encapsulated key.

Our Contribution. There has been a natural trend in research community to extend any advanced cryptographic primitive from identity based to more flexible and practical attribute based framework. In this paper, we introduce a refined version of *revocable ciphertext policy attribute based key encapsulation mechanism* (RCP-ABKEM) motivated by *key-homomorphic* IRKEM of Sun et al. (Sun et al., 2020). We define *extended correctness* and *key-homomorphism* in attribute based setting. The new primitive with these additional properties can be plugged into various privacy preserving protocols. However, this realization is non-trivial. The challenge lies in the requirement of introducing the additional properties in an RCP-ABKEM. We provide an instantiation of RCP-ABKEM which is proven to be selectively secure against *chosen plaintext attack* (CPA). The underlying hardness of our construction is *q*-DBDHE problem and security proof of our scheme is in the standard model. Our design differs from the work of Liu et al. (Liu et al., 2018) in the sense that we do not consider secret key time validation technique. Our security model allows an adversary to ask secret key of a user whose identity is in the revocation list or whose attribute set does not satisfy the challenge access structure. We briefly summarize the comparison of communication banwidth, storage and other functionality of our scheme in reference to the existing work of Balu and Kuppusamy (Balu and Kuppusamy, 2013) and Wang et al. (Wang et al., 2017) in Table 1. Similar to the work of (Balu and Kuppusamy, 2013) and (Wang et al., 2017), we use symmetric bilinear pairing $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_1$ where $\mathbb{G}$, $\mathbb{G}_1$ are cyclic groups of prime order p. We emphasize that the master secret key size ($|\mathsf{msk}|$) of our scheme is less than those in (Balu and Kuppusamy, 2013) and (Wang et al., 2017). Moreover, our ciphertext size is significantly shorter than those in (Balu and Kuppusamy, 2013) and (Wang et al., 2017). The ciphertext size in our construction is $(l+2)|\mathbb{G}|$ which does not depend on number of revoked users r while that of (Balu and Kuppusamy, 2013), (Wang et al., 2017) are respectively $(1+2r+l)|\mathbb{G}|$, $(1+2lr)|\mathbb{G}|$ which grows as the number of revoked user grows. Here l is number of attributes present in the access structure $\Gamma = (N, \eta)$. Our public key size is $|\mathsf{pk}| = (n_{\mathsf{rev}} + n_{\mathsf{att}} + 2)|\mathbb{G}| + |\mathbb{G}_1|$ which includes public parameter size $|\mathsf{pp}| = (n_{\mathsf{rev}} + n_{\mathsf{att}} + 2)|\mathbb{G}|$ and master public key size $|\mathsf{mpk}| = |\mathbb{G}_1|$ where $(n_{\mathsf{rev}} - 1)$ denotes maximum number of revoked users in the system and n_{att} is the cardinality of the attribute space associated with system. However, size of public key ($|\mathsf{pk}|$) and secret key $|\mathsf{sk}|$ are more in our design as compared to that of (Balu and Kuppusamy, 2013) and

Table 1: Comparison of communication bandwidth, storage overhead and key-homomorphic property.

Scheme	Communication	Storage			KH
	$\|\mathsf{ct}\|$	$\|\mathsf{pk}\|$	$\|\mathsf{msk}\|$	$\|\mathsf{sk}\|$	
(Balu and Kuppusamy, 2013)	$(1+2r+l)\|\mathbb{G}\|$	$(n_{\mathsf{att}}+2)\|\mathbb{G}\|+\|\mathbb{G}_1\|$	$(n_{\mathsf{att}}+2)\|\mathbb{Z}_p\|$	$(\|S\|+3)\|\mathbb{G}\|$	No
(Wang et al., 2017)	$(1+2lr)\|\mathbb{G}\|$	$(n_{\mathsf{att}}+3)\|\mathbb{G}\|+\|\mathbb{G}_1\|$	$2\|\mathbb{Z}_p\|$	$(\|S\|+2)\|\mathbb{G}\|$	No
Our	$(l+2)\|\mathbb{G}\|$	$(n_{\mathsf{rev}}+n_{\mathsf{att}}+2)\|\mathbb{G}\|+\|\mathbb{G}_1\|$	$\|\mathbb{Z}_p\|$	$(n_{\mathsf{rev}}+\|S\|+2)\|\mathbb{G}\|$	Yes

$|\mathsf{ct}|$ = ciphertext size, $|\mathsf{pk}|$ = public key size, $|\mathsf{msk}|$ = master secret key size, $|\mathsf{sk}|$ = secret key size of a user with identity id and attribute set S, KH = Key-homomorphic, HA = hardness assumption, $|\mathbb{G}|$ = bit size of an element of $\mathbb{G}$, $|\mathbb{G}_1|$ = bit size of an element of $\mathbb{G}_1$, l = number of attributes present in the access structure, r = number of revoked users specified at encryption phase, n_{att} = the cardinality of the attribute space $\mathcal{AS}$ associated with system, $(n_{\mathsf{rev}}-1)$ = maximum number of revoked users in our system

(Wang et al., 2017). With this tradeoff, we achieve the first revocable encapsulation scheme preserving *key-homomorphic* property in attribute based setting.

A Sample Application. RCP-ABKEM with *key-homomorphism* property and *extended correctness* can enable numerous powerful cryptographic application. For instance, we can extend the notion of PKEM in attribute based setting and employ the *key-homomorphism* and *extended correctness* of RCP-ABKEM to puncture a set of attributes using tag based approach following (Sun et al., 2020). Precisely, this variant of PKEM in attribute based setting consists of a tuple (KeyGen, Puncture, Encaps, Decaps). A user Alice runs the algorithm KeyGen to generate public key PK and secret key SK. She publishes PK and keeps SK secret to herself. Alice runs the algorithm Puncture taking input a tag t which is an attribute set, secret key SK_{i-1} where $\mathsf{SK}_0 = \mathsf{SK}$ and generates a new secret key SK_i. An encryptor executes Encaps on input public key PK and a list of tags T consisting of access structures to generate encapsulated key K and ciphertext CT. On input secret key SK_i, ciphertext CT along with tag list T, Alice performs the algorithm Decaps to recover the encapsulated key K. The correctness requirement is that SK_i can decapsulate all the ciphertexts decapsulated by SK_{i-1} except for those encrypted under tag t. In other words, for any l times of invoking $\mathsf{SK}_i \leftarrow \mathsf{Punc}(\mathsf{SK}_{i-1}, t)$ such that t does not satisfy any of the access structures in T, decapsulation will be successful with overwhelming probability on input $\mathsf{SK}_l, \mathsf{CT}, T$. For example, consider a scenario where Alice has applied for mail subscription to a group of companies for job updates where each company owns a set of attributes. Suppose it has been exposed that "companies which are private and foreign" sends malware to recipients. For safety, Alice can ultilize Puncture on tag S ={"foreign", "private"} to generate a new secret key so that using *key-homomorphism* and *extended correctness*, she can decapsulate only those ciphertexts CT with associated tag list T such that S does not satisfy any of the access structures in the tag list T. Again if there is any threat regarding tag S' = {"sports medicine","multinational"}, she can again Puncture on tag S' so that she can decapsulate only those ciphertexts CT with associated tag list T such that S, S' satisfy none of the access structures in the tag list T.

2 PRELIMINARIES

2.1 Notation

Let λ denotes the security parameter. By $x \xleftarrow{u} S$ we mean that x is chosen uniformly from the set S. Let $\varnothing$ stands for the empty set and $[n]$ represents the set $\{1,2,\ldots,n\}$ for any $n \in \mathbb{N}$. We say $f : \mathbb{N} \to \mathbb{R}$ is a *negligible* function of n if it is $O(n^{-c})$ for all $c > 0$ and we use $\mathsf{negl}(n)$ for negligible function of n. Let the symbol $\perp$ indicates failure or null value, $\mathscr{P}(A)$ denotes the power set of a set A and $\langle \cdot, \cdot \rangle$ represents inner product of two vectors.

2.2 Access Structure

Let $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ be a set of parties. A collection $\Gamma \subseteq \mathscr{P}(\mathcal{P})$ is monotone if for any sets B, C satisfying $B \in \Gamma, B \subseteq C \subseteq \mathcal{P}$, we have $C \in \Gamma$. A monotone access structure on $\mathcal{P}$ is a monotone collection Γ of non-empty subsets of $\mathcal{P}$, that is $\Gamma \subseteq \mathscr{P}(\mathcal{P}) \setminus \varnothing$. The sets in Γ are called the authorized sets and the sets in $\mathscr{P}(\mathcal{P}) \setminus \Gamma$ are called the unauthorized sets. We say that a set S satisfies the access structure Γ if S is an authorized set i.e. $S \in \Gamma$. Unless otherwise stated, by an access structure we indicate monotone access structure throughout this paper.

2.3 Linear Secret Sharing Scheme

A secret sharing scheme Λ over a set of parties $\mathcal{P}$ is called *linear* over $\mathbb{Z}_p$ if

1. The shares of each party form a vector over $\mathbb{Z}_p$.
2. There is a matrix N having l rows and d columns called the share generating matrix for Λ and a

function η that associates each row of the matrix N to a corresponding party. For $i \in [l]$, the i-th row N_i of N is labeled by a party $\eta(i)$ where η is a function from $[l]$ to $\mathcal{P}$. Given the column vector $\mathbf{v} = (s, z_2, \ldots, z_d)$ where s is the secret to be shared and $z_2, z_3, \ldots, z_d$ are chosen uniformly at random, $N\mathbf{v}$ is the vector of l shares of the secret s according to Λ. The share $\lambda_i = (N\mathbf{v})_i = \langle N_i, \mathbf{v} \rangle$ belongs to party $\eta(i)$.

As exhibited by Beimel et al. (Beimel et al., 1996), any *linear secret sharing scheme* (LSSS) satisfies the linear reconstruction property stated in the following Theorem:

Theorem 1. *Let $\Lambda = (N, \eta)$ be an* LSSS *for access structure Γ and $S \in \Gamma$ be an authorized set. Let $I \subset [l]$ be defined as $I = \{i : \eta(i) \in S\}$. Then there exists constants $\{w_i \in \mathbb{Z}_p\}_{i \in I}$ satisfying $\sum_{i \in I} w_i N_i = (1, 0, \ldots, 0)$ so that when λ_i are valid shares of a secret s according to Λ, $\sum_{i \in I} w_i \lambda_i = s$. Moreover, these constants $\{w_i \in \mathbb{Z}_p\}_{i \in I}$ can be computed in time polynomial in the size of the share generating matrix N.*

2.4 Revocable Ciphertext-policy Attribute-based Key Encapsulation Mechanism with Key-homomorphic Property

In this section, we recall the syntax and security of *revocable ciphertext-policy attribute based key encapsulation mechanism* (RCP-ABKEM) from (Liu and Wong, 2015) and introduce a new concept – *key-homomorphic* property in RCP-ABKEM. An RCP-ABKEM scheme with master public key space $\mathcal{MPK}$, master secret key space $\mathcal{MSK}$, secret key space $\mathcal{SK}$, encapsulated key space $\mathcal{EK}$, identity space $\mathcal{ID}$, attribute space $\mathcal{AS}$ and randomness spaces $\mathcal{RS}, \mathcal{RS}'$ consists of a tuple of polynomial time algorithms RCP-ABKEM=(Setup, MKGen, KeyExt, Encaps, Decaps) satisfying the following requirements.

- $\mathsf{Setup}(1^\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}}) \rightarrow \mathsf{pp}$: A trusted authority runs this probabilistic algorithm on input the security parameter λ, number of attributes $n_{\mathsf{att}} \in \mathbb{N}$, an integer $n_{\mathsf{rev}} \in \mathbb{N}$ and generates public parameter pp. Here $(n_{\mathsf{rev}} - 1)$ is the maximum number of revoked users is in the system.
- $\mathsf{MKGen}(\mathsf{pp}) \rightarrow (\mathsf{mpk}, \mathsf{msk})$: The trusted authority takes input the public parameter pp and probabilistically outputs a master public key $\mathsf{mpk} \in \mathcal{MPK}$ and a master secret key $\mathsf{msk} \in \mathcal{MSK}$. It publishes mpk and keeps msk secret to itself.
- $\mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}, \mathsf{id}, S ; \tau) \rightarrow \mathsf{sk}_{\mathsf{id},S}$: On input the public parameter pp, a user identity $\mathsf{id} \in \mathcal{ID}$ with attribute set $S \subseteq \mathcal{AS}$, the trusted authority chooses randomness $\tau \xleftarrow{u} \mathcal{RS}'$ and runs this algorithm to generate a private key $\mathsf{sk}_{\mathsf{id},S} \in \mathcal{SK}$ using the master secret key msk. It issues $\mathsf{sk}_{\mathsf{id},S}$ to the identity holder id with attribute set S through a secure channel between them.
- $\mathsf{Encaps}(\mathsf{pp}, \mathsf{mpk}, \mathcal{R}, \Gamma; \{r_i\}_{i \in [n]}) \rightarrow (k, \mathsf{ct}, \mathcal{R}, \Gamma)$: An encapsulator takes as input the public parameter pp, the master public key mpk, a list $\mathcal{R} \subseteq \mathcal{ID}$ of revoked users identity with $|\mathcal{R}| < n_{\mathsf{rev}}$, an access structure Γ over the attribute space $\mathcal{AS}$ and probabilistically outputs a symmetric key k and a ciphertext ct. It publishes ct along with description of $\mathcal{R}, \Gamma$ and keeps k secret to itself. Here $r_i \in \mathcal{RS}$ for $i \in [n]$ are randomness used in the algorithm where n is a positive integer.
- $\mathsf{Decaps}(\mathsf{pp}, \mathsf{sk}_{\mathsf{id},S}, \mathsf{ct}, \mathcal{R}, \Gamma) \rightarrow k / \perp$: On input the public parameter pp, a private key $\mathsf{sk}_{\mathsf{id},S}$ for an identity id with attribute set S and a ciphertext ct associated with the revocation list $\mathcal{R}$ and access structure Γ over the attribute space $\mathcal{AS}$, a decapsulator recovers the encapsulated key k or $\perp$.

Correctness. For correctness, we require that any user with private key $\mathsf{sk}_{\mathsf{id},S}$ for $\mathsf{id} \notin \mathcal{R}, S \in \Gamma$ can retrieve the encapsulated key k. More formally, an RCP-ABKEM scheme is said to be correct if for all $\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}} \in \mathbb{N}$, $\mathcal{R} \subseteq \mathcal{ID}$ with $|\mathcal{R}| < n_{\mathsf{rev}}$, $\Gamma \subseteq \mathscr{P}(\mathcal{AS}) \setminus \varnothing$ with $|\mathcal{AS}| = n_{\mathsf{att}}$, $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}})$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{MKGen}(\mathsf{pp})$, $(k, \mathsf{ct}, \mathcal{R}, \Gamma) \leftarrow \mathsf{Encaps}(\mathsf{pp}, \mathsf{mpk}, \mathcal{R}, \Gamma; \{r_i\}_{i \in [n]})$ and $\mathsf{sk}_{\mathsf{id},S} \leftarrow \mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}, \mathsf{id}, S; \tau)$ for $\mathsf{id} \notin \mathcal{R}, S \in \Gamma$, it holds that

$$\Pr[\mathsf{Decaps}(\mathsf{pp}, \mathsf{sk}_{\mathsf{id},S}, \mathsf{ct}, \mathcal{R}, \Gamma) = k] \geq 1 - \mathsf{negl}(\lambda)$$

Security. We describe the *selective security* model of RCP-ABKEM against *chosen plaintext attack* (CPA) following the security model of Liu et al. (Liu and Wong, 2015). The game played between an adversary $\mathcal{B}$ and a challenger $\mathcal{C}$ is detailed below.

- **Init:** The adversary $\mathcal{B}$ submits the challenge revocation list $\mathcal{R}^\star$ and the challenge access structure $\Gamma^\star$ to the challenger $\mathcal{C}$.
- **Setup:** The challenger $\mathcal{C}$ generates $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}})$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{MKGen}(\mathsf{pp})$ and provides pp, mpk to $\mathcal{B}$.
- **Query Phase 1:** The adversary makes polynomially many private key queries. The challenger $\mathcal{C}$ responds with $\mathsf{sk}_{\mathsf{id},S} \leftarrow \mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}, \mathsf{id}, S; \tau)$ to $\mathcal{B}$ corresponding to the query on the identity id and the attribute set S if $\mathsf{id} \in \mathcal{R}^\star$ or $S \notin \Gamma^\star$.

- **Challenge:** The challenger $\mathcal{C}$ flips a random coin $b \in \{0,1\}$, selects $k_1^\star \xleftarrow{u} \mathcal{EK}$ and computes $(k_0^\star, \mathsf{ct}^\star, \mathcal{R}^\star, \Gamma^\star) \leftarrow \mathsf{Encaps}(\mathsf{pp}, \mathsf{mpk}, \mathcal{R}^\star, \Gamma^\star; \{r_i\}_{i\in[n^\star]})$ where $n^\star \in \mathbb{N}$. The challenger $\mathcal{C}$ sends $(k_b^\star, \mathsf{ct}^\star)$ to the adversary $\mathcal{B}$.
- **Query Phase 2:** Same as phase 1.
- **Guess:** At the end, $\mathcal{B}$ outputs a guess bit b' and wins the game if $b' = b$.

Advantage of $\mathcal{B}$ in this game is defined as:

$$\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{RCP-ABKEM},\mathcal{B}}(\lambda) = \left|\mathsf{Pr}[b' = b] - \frac{1}{2}\right|$$

We say that an RCP-ABKEM scheme is CPA-secure if $\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{RCP-ABKEM},\mathcal{B}}(\lambda) \leq \mathsf{negl}(\lambda)$.

Extended Correctness. An RCP-ABKEM is said to satisfy extended correctness if the following holds. Let $\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}} \in \mathbb{N}$, for $i \in \{1,2\}$ let $\mathcal{R}_i \subseteq \mathcal{ID}$ with $|\mathcal{R}_i| \leq n_{\mathsf{rev}}$, $\Gamma_i \subseteq \mathscr{P}(\mathcal{AS}) \setminus \varnothing$ with $|\mathcal{AS}| = n_{\mathsf{att}}$, $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}})$, $(\mathsf{mpk}_i, \mathsf{msk}_i) \leftarrow \mathsf{MKGen}(\mathsf{pp})$, $(k_i, \mathsf{ct}_i, \mathcal{R}_i, \Gamma_i) \leftarrow \mathsf{Encaps}(\mathsf{pp}, \mathsf{mpk}_i, \mathcal{R}_i, \Gamma_i; \{r_{ij}\}_{j\in[n_i]})$ where $n_i \in \mathbb{N}$, $\{r_{ij}\}_{j\in[n]}$ is a set of randomness used in encapsulation algorithm, $\mathsf{sk}_i \leftarrow \mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}_i, \mathsf{id}_i, S_i; \tau_i)$, $(\widehat{k}, \widehat{\mathsf{ct}}, \mathcal{R}_2, \Gamma_2) \leftarrow \mathsf{Encaps}(\mathsf{pp}, \mathsf{mpk}_1, \mathcal{R}_2, \Gamma_2; \{r_{2j}\}_{j\in[n_2]})$. Then for $\mathsf{id}_1 \notin \mathcal{R}_2, S_1 \in \Gamma_2$ it holds that $\mathsf{Pr}[\mathsf{Decaps}(\mathsf{pp}, \mathsf{sk}_1 = \mathsf{sk}_{\mathsf{id}_1,S_1}, \mathsf{ct}_2, \mathcal{R}_2, \Gamma_2) = \widehat{k}] \geq 1 - \mathsf{negl}(\lambda)$.

Key-homomorphism. We let that the randomness space $\mathcal{RS}'$ (associated with KeyExt), master secret key space $\mathcal{MSK}$, secret key space $\mathcal{SK}$ and encapsulated key space $\mathcal{EK}$ of an RCP-ABKEM scheme form four groups $(\mathcal{RS}', *)$, $(\mathcal{MSK}, +)$, $(\mathcal{SK}, \otimes)$, $(\mathcal{EK}, \odot)$. Furthermore, we assume that the encapsulated key k is of the form $f(\mathsf{msk}, s)$ where s is one of the randomness used in encapsulation algorithm. Then the RCP-ABKEM scheme is said to satisfy key-homomorphic property if extended correctness holds and for all $\mathsf{id} \in \mathcal{ID}$, msk, $\mathsf{msk}' \in \mathcal{MSK}$, $\tau, \tau' \in \mathcal{RS}'$ the following two conditions hold.

1. $\mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}, \mathsf{id}, S; \tau) \otimes \mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}', \mathsf{id}, S; \tau') = \mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk} + \mathsf{msk}', \mathsf{id}, S; \tau * \tau')$.
2. $f(\mathsf{msk}, s) \odot f(\mathsf{msk}', s) = f(\mathsf{msk} + \mathsf{msk}', s)$.

2.5 Symmetric Bilinear Map and Hardness Assumption

Definition 2.5.1. (Symmetric Bilinear Map). *Let $\mathbb{G}, \mathbb{G}_1$ be multiplicative cyclic groups of prime order p and Let g be a generator of $\mathbb{G}$. A symmetric bilinear mapping $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ is a function having the following properties.*

1. *$e(u^a, v^b) = e(u,v)^{ab}$ $\forall\, u, v \in \mathbb{G}$ and $\forall a, b \in \mathbb{Z}_p$.*
2. *The map is non degenerate, i.e., $e(g,g)$ is a generator of $\mathbb{G}_1$.*

The tuple $(p, \mathbb{G}, \mathbb{G}_1, e)$ is called a prime order symmetric bilinear group system.

Definition 2.5.2. (q-Decisional Bilinear Diffie-Hellman Exponent (DBDHE) Problem). *Let $\mathbb{G}, \mathbb{G}_1$ be cyclic groups of prime order p and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ be a symmetric bilinear pairing generated by a bilinear group generator on input a security parameter λ. Let $s, b \xleftarrow{u} \mathbb{Z}_p$, g be a random generator of $\mathbb{G}$, $z_1 = (g, g^s, g^b, g^{b^2}, \ldots, g^{b^q}, g^{b^{q+2}}, \ldots, g^{b^{2q}})$ and $z_2 \in \mathbb{G}_1$. Given (z_1, z_2), the q-DBDHE problem is to determine whether z_2 is $e(g,g)^{sb^{q+1}}$ or a random element of $\mathbb{G}_1$. The advantage of a distinguisher $\mathcal{B}$ is defined as $\mathsf{Adv}^{q-\mathsf{DBDHE}}_{\mathcal{B}}(\lambda) = |\mathsf{Pr}[\mathcal{B}(z_1, e(g,g)^{sb^{q+1}}) \to 1] - \mathsf{Pr}[\mathcal{B}(z_1, z_2) \to 1]|$.*

3 CONSTRUCTION OF RCP-ABKEM FEATURING KEY-HOMOMORPHIC PROPERTY

Our construction of key-homomorphic revocable ciphertext-policy attribute-based key encapsulation mechanism RCP-ABKEM = (Setup, MKGen, KeyExt, Encaps, Decaps) uses bilinear setup and has identity space $\mathcal{ID} = \mathbb{Z}_p$, master public key space $\mathcal{MPK} = \mathbb{G}_1$, master secret key space $\mathcal{MSK} = \mathbb{Z}_p$, secret key space $\mathcal{SK} = \mathbb{G}^n$ for some $n \in \mathbb{N}$, randomness space $\mathcal{RS} = \mathbb{Z}_p, \mathcal{RS}' = \mathbb{Z}_p^2$ and encapsulated key space $\mathcal{EK} = \mathbb{G}_1$, where $\mathbb{G}$, $\mathbb{G}_1$ are two cyclic groups of prime order p as defined in Def. 2.5.1 above. The scheme is described below.

- $\mathsf{Setup}(1^\lambda, n_{\mathsf{att}}, n_{\mathsf{rev}}) \to \mathsf{pp}$: A trusted authority takes as input the security parameter λ, two positive integers n_{att} and n_{rev} where n_{att} is the number of attributes, $(n_{\mathsf{rev}} - 1)$ is the maximum number of revoked users in the system. It generates a pair of bilinear groups $\mathbb{G}$, $\mathbb{G}_1$ of prime order $p > 2^\lambda$ with bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$. It randomly picks $g, p_1, p_2, \ldots, p_{n_{\mathsf{att}}} \in \mathbb{G}$, $\gamma_0 \in \mathbb{Z}_p, \boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_{n_{\mathsf{rev}}})^\top \in \mathbb{Z}_p^{n_{\mathsf{rev}}}$, computes $g^{\gamma_0}, c_i = g^{\gamma_i}$ for $i \in [n_{\mathsf{rev}}]$ and publishes

$$\mathsf{pp} = \left(g, g^{\gamma_0}, c_1, c_2, \ldots, c_{n_{\mathsf{rev}}}, p_1, p_2, \ldots, p_{n_{\mathsf{att}}}\right)$$

- $\mathsf{MKGen}(\mathsf{pp}) \to (\mathsf{mpk}, \mathsf{msk})$: On input the public parameter pp, the trusted authority randomly chooses $\alpha \in \mathbb{Z}_p$, sets master secret key $\mathsf{msk} = \alpha \in \mathbb{Z}_p$ and master public key $\mathsf{mpk} = e(g,g)^\alpha \in \mathbb{G}_1$. It publishes mpk and keeps msk secret to itself.

- $\mathsf{KeyExt}(\mathsf{pp}, \mathsf{msk}, \mathsf{id}, S; \tau = (u,a)) \rightarrow \mathsf{sk}_{\mathsf{id},S}$: Taking input the public parameter pp, the master secret key $\mathsf{msk} = \alpha$, attribute set $S \subseteq \mathcal{AS}$ of a user with identity $\mathsf{id} \in \mathcal{ID}$, the trusted authority chooses $(u,a) \in \mathbb{Z}_p^2$ at random and computes $\mathsf{sk}_{\mathsf{id},S}$ as $\mathsf{sk}_{\mathsf{id},S} = \Big(\psi_0 = g^a, \psi_0' = g^u, \psi_1 = g^{\alpha} g^{\gamma_0 a} g^{\gamma_1 u}, \{h_x = p_x^a\}_{x \in S}, \{\delta_i = (c_1{}^{-\mathsf{id}^{i-1}} c_i)^u\}_{i=2}^{n_{\mathsf{rev}}}\Big)$.

  It issues the secret key $\mathsf{sk}_{\mathsf{id},S}$ to the identity holder id with attribute set S through a secure channel between them.

- $\mathsf{Encaps}(\mathsf{pp}, \mathsf{mpk}, \mathcal{R}, \Gamma; \{s, z_2, \ldots, z_d\}) \rightarrow (k, \mathsf{ct}, \mathcal{R}, \Gamma)$: Given the public parameter $\mathsf{pp} = \Big(g, g^{\gamma_0}, c_1, c_2, \ldots, c_{n_{\mathsf{rev}}}, p_1, p_2, \ldots, p_{n_{\mathsf{att}}}\Big)$, master public key $\mathsf{mpk} = e(g,g)^{\alpha}$, a revocation list $\mathcal{R} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_r\}$ for some positive integer $r < n_{\mathsf{rev}}$, an LSSS access structure $\Gamma = (N, \eta)$ where N is an $l \times d$ matrix over $\mathbb{Z}_p$ and η associates rows of N to attributes, an encapsulator randomly chooses a vector $\mathbf{v} = (s, z_2, \ldots, z_d) \in \mathbb{Z}_p^d$, computes share $\lambda_i = \langle \mathbf{v}, N_i \rangle$ of s for $i \in [l]$ where N_i denotes the i-th row of N and generates the encapsulated key k and ciphertext $\mathsf{ct} = (C_0', C_0'', C_1, C_2, \ldots, C_l)$ by executing the following steps:

  1. Defines a polynomial $f_{\mathcal{R}}(Z) = (Z - \mathsf{id}_1)(Z - \mathsf{id}_2) \cdots (Z - \mathsf{id}_r) = y_1 + y_2 Z + \cdots + y_r Z^{r-1} + y_{r+1} Z^r$ and sets $y_{r+2} = y_{r+3} = \ldots = y_{n_{\mathsf{rev}}} = 0$
  2. Calculates $k = (e(g,g)^{\alpha})^s$ and sets $\mathsf{ct} = (C_0', C_0'', C_1, \ldots, C_l)$ with $C_0' = g^s, C_0'' = (\prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i})^s, C_1 = g^{\gamma_0 \lambda_1} p_{\eta(1)}^{-s}, \ldots, C_l = g^{\gamma_0 \lambda_l} p_{\eta(l)}^{-s}$ where $\mathsf{mpk} = e(g,g)^{\alpha}$ and $g, c_1, c_2, \ldots, c_{n_{\mathsf{rev}}}, g^{\gamma_0}, p_{\eta(1)}, p_{\eta(2)}, \ldots, p_{\eta(l)}$ are extracted from pp. Note that $\{\eta(1), \eta(2), \ldots, \eta(l)\} \subset \{1, 2, \ldots, n_{\mathsf{att}}\}$.

  It publishes ct along with the description of $\mathcal{R}$, Γ and keeps k secret to itself.

- $\mathsf{Decaps}(\mathsf{pp}, \mathsf{sk}_{\mathsf{id},S}, \mathsf{ct}, \mathcal{R}, \Gamma) \rightarrow k / \perp$: On input the public parameter pp, a private key $\mathsf{sk}_{\mathsf{id},S}$ for an identity id with attribute set S and a ciphertext ct $= (C_0', C_0'', C_1, \ldots, C_l)$ under the revocation list $\mathcal{R}$ and access structure $\Gamma = (N, \eta)$, a decapsulator proceeds as follows:

  1. Defines a vector $\mathbf{x} = (1, \mathsf{id}, \mathsf{id}^2, \ldots, \mathsf{id}^{n_{\mathsf{rev}}-1})$ from the identity id and a vector $\mathbf{y} = (y_1, y_2, \ldots, y_{n_{\mathsf{rev}}})$ from the revoked list $\mathcal{R} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_r\}$ where y_i, $i \in [r+1]$, are coefficients of Z^{i-1} in the polynomial $f_{\mathcal{R}}(Z) = \prod_{i=1}^{r} (Z - \mathsf{id}_i) = \sum_{i=1}^{r+1} y_i Z^{i-1}$ and $y_{r+2} = y_{r+3} = \ldots = y_{n_{\mathsf{rev}}} = 0$ if $r+1 < n_{\mathsf{rev}}$.
  2. Returns $\perp$ if either $\mathsf{id} \in \mathcal{R}$ i.e. $f_{\mathcal{R}}(\mathsf{id}) = \langle \mathbf{x}, \mathbf{y} \rangle = 0$ or S does not satisfy the access structure $\Gamma = (N, \eta)$.
  3. Otherwise, computes
  $$C = \prod_{i=2}^{n_{\mathsf{rev}}} \delta_i^{y_i}$$
  and
  $$\kappa_1 = \left(\frac{e(C, C_0')}{e(\psi_0', C_0'')} \right)^{\frac{-1}{\langle \mathbf{x}, \mathbf{y} \rangle}}$$
  by extracting ψ_0' and δ_i, $i = 2, \ldots, n_{\mathsf{rev}}$ from $\mathsf{sk}_{\mathsf{id},S}$ and C_0', C_0'' from ct.
  4. Defines $I = \{i \in [l] : \eta(i) \in S\} \subset [l]$, finds a set of constants $\{w_i \in \mathbb{Z}_p\}_{i \in I}$ satisfying $\sum_{i \in I} w_i \lambda_i = s$ where $\lambda_i = \langle N_i, \mathbf{v} \rangle$ are valid shares of the secret s according to $\Gamma = (N, \eta)$, N_i being the i-th row of the matrix N, s being the secret randomness picked to compute ct during encapsulation and $\mathbf{v} = (s, z_2, \ldots, z_d) \in \mathbb{Z}_p^d$. Since S satisfies the access structure $\Gamma = (N, \eta)$, the set of constants $\{w_i \in \mathbb{Z}_p\}_{i \in I}$ can be computed in polynomial time by Theorem 1.
  5. Computes
  $$\kappa_2 = \prod_{i \in I} \left(e(C_i, \psi_0) e(C_0', h_{\eta(i)}) \right)^{w_i}$$
  and recovers
  $$k = \frac{e(\psi_1, C_0')}{\kappa_1 \kappa_2}$$
  where ψ_0, ψ_1 and $h_{\eta(i)}, i \in I$ are extracted from $\mathsf{sk}_{\mathsf{id},S}$ and $C_0', C_i, i \in I$ are obtained from ct.

Table 2: Computation cost of our scheme.

Algorithm	# exp	# pair
Setup	$(n_{\mathsf{rev}} + 1)\lvert\mathbb{G}\rvert$	0
MKGen	$\lvert\mathbb{G}_1\rvert$	1
KeyExt	$(2n_{\mathsf{rev}} + \lvert S\rvert + 3)\lvert\mathbb{G}\rvert$	0
Encaps	$(n_{\mathsf{rev}} + 2l + 1)\lvert\mathbb{G}\rvert + \lvert\mathbb{G}_1\rvert$	0
Decaps	$(n_{\mathsf{rev}} - 1)\lvert\mathbb{G}\rvert + \lvert I\rvert\lvert\mathbb{G}_1\rvert$	$(3 + 2\lvert I\rvert)$

Correctness. The correctness follows as

$$\begin{aligned} C &= \prod_{i=2}^{n_{\mathsf{rev}}} \delta_i^{y_i} = \prod_{i=2}^{n_{\mathsf{rev}}} \left(c_1{}^{-\mathsf{id}^{i-1}} c_i \right)^{u y_i} \\ &= \left(c_1{}^{-\langle \mathbf{x}, \mathbf{y} \rangle + y_1} \prod_{i=2}^{n_{\mathsf{rev}}} c_i^{y_i} \right)^u \\ &= \left(c_1{}^{-\langle \mathbf{x}, \mathbf{y} \rangle} \prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i} \right)^u \end{aligned}$$

$$\begin{aligned}
\kappa_1 &= \left(\frac{e(C,C_0')}{e(\psi_0',C_0'')}\right)^{\frac{-1}{\langle \mathbf{x},\mathbf{y}\rangle}} \\
&= \left(\frac{e((c_1{}^{-\langle \mathbf{x},\mathbf{y}\rangle}\prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i})^u, g^s)}{e(g^u,(\prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i})^s)}\right)^{\frac{-1}{\langle \mathbf{x},\mathbf{y}\rangle}} \\
&= \left(\frac{e(c_1{}^{-u\langle \mathbf{x},\mathbf{y}\rangle}, g^s)e((\prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i})^u, g^s)}{e(g^u,(\prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i})^s)}\right)^{\frac{-1}{\langle \mathbf{x},\mathbf{y}\rangle}} \\
&= \left(e(c_1{}^{-u\langle \mathbf{x},\mathbf{y}\rangle}, g^s)\right)^{\frac{-1}{\langle \mathbf{x},\mathbf{y}\rangle}} \\
&= e(c_1^u, g^s) = e(g,g)^{\gamma_1 su}
\end{aligned}$$

$$\begin{aligned}
\kappa_2 &= \prod_{i\in I}\left(e(C_i,\psi_0)e(C_0',h_{\eta(i)})\right)^{w_i} \\
&= \prod_{i\in I}\left(e(g^{\gamma_0\lambda_i}p_{\eta(i)}^{-s}, g^a)e(g^s, p_{\eta(i)}^a)\right)^{w_i} \\
&= \prod_{i\in I}\left(e(g,g)^{\gamma_o\lambda_i a}e(p_{\eta(i)},g)^{-sa}e(g,p_{\eta(i)})^{sa}\right)^{w_i} \\
&= \prod_{i\in I} e(g,g)^{\gamma_0\lambda_i a w_i} \\
&= e(g,g)^{\gamma_0 a\sum_{i\in I} w_i\lambda_i} = e(g,g)^{\gamma_0 as}
\end{aligned}$$

Therefore,

$$\begin{aligned}
k &= \frac{e(\psi_1,C_0')}{\kappa_1\kappa_2} = \frac{e(g^\alpha g^{\gamma_0 a}g^{\gamma_1 u}, g^s)}{e(g,g)^{\gamma_1 su}e(g,g)^{\gamma_0 as}} \\
&= \frac{e(g^\alpha,g^s)e(g^{\gamma_0 a},g^s)e(g^{\gamma_1 u},g^s)}{e(g,g)^{\gamma_1 su}e(g,g)^{\gamma_0 as}} = e(g,g)^{\alpha s}
\end{aligned}$$

Extended Correctness. Let $(\mathsf{mpk}_0,\mathsf{msk}_0) = (e(g,g)^{\alpha_0},\alpha_0) \leftarrow \mathsf{MKGen}(\mathsf{pp})$ be another master public and secret key pair and $\mathsf{sk}_{\mathsf{id}_0,S_0} = (\zeta_0 = g^{a_0}, \zeta_0' = g^{u_0}, \zeta_1 = g^{\alpha_0}g^{\gamma_0 a_0}g^{\gamma_1 u_0}, \{h_x' = p_x^{a_0}\}_{x\in S_0}, \{\delta_i' = (c_1{}^{-\mathsf{id}_0^{i-1}}c_i)^{u_0}\}_{i=2}^{n_{\mathsf{rev}}})$ generated by running $\mathsf{KeyExt}(\mathsf{pp},\mathsf{msk}_0,\mathsf{id}_0,S_0;\tau_0 = (u_0,a_0))$ be a secret key corresponding to the identity $\mathsf{id}_0 \in \mathbb{Z}_p$ with attribute set $S_0 \subseteq \mathcal{AS}$ where $(u_0,a_0) \xleftarrow{u} \mathbb{Z}_p^2$. We observe that if $\mathsf{id}_0 \notin \mathcal{R}$ and $S_0 \in \Gamma = (N,\eta)$ then $\mathsf{Decaps}(\mathsf{pp},\mathsf{sk}_{\mathsf{id}_0,S_0},\mathsf{ct},\mathcal{R},\Gamma) = \widehat{k} = e(g,g)^{\alpha_0 s}$ as follows where $(\widehat{k},\widehat{\mathsf{ct}},\mathcal{R},\Gamma) \leftarrow \mathsf{Encaps}(\mathsf{pp},\mathsf{mpk}_0,\mathcal{R},\Gamma;\{s,z_2,\ldots,z_d\})$ and $(k,\mathsf{ct},\mathcal{R},\Gamma) \leftarrow \mathsf{Encaps}(\mathsf{pp},\mathsf{mpk},\mathcal{R},\Gamma;\{s,z_2,\ldots,z_d\})$.

1. A decapsulator with secret key $\mathsf{sk}_{\mathsf{id}_0,S_0}$ defines a vector $\mathbf{x}_0 = (1,\mathsf{id}_0,\mathsf{id}_0^2,\ldots,\mathsf{id}_0^{n_{\mathsf{rev}}-1})$ from the identity id_0 and a vector $\mathbf{y} = (y_1,y_2,\ldots,y_{n_{\mathsf{rev}}})$ from the revoked list $\mathcal{R} = \{\mathsf{id}_1,\mathsf{id}_2,\ldots,\mathsf{id}_r\}$ where y_i, $i \in [r+1]$, are coefficients of Z^{i-1} in the polynomial $f_{\mathcal{R}}(Z) = \prod_{i=1}^{r}(Z-\mathsf{id}_i) = \sum_{i=1}^{r+1} y_i Z^{i-1}$ and $y_{r+2} = y_{r+3} = \ldots = y_{n_{\mathsf{rev}}} = 0$ if $r+1 < n_{\mathsf{rev}}$. As $\mathsf{id}_0 \notin \mathcal{R}$, $\langle \mathbf{x}_0,\mathbf{y}\rangle \neq 0$. It computes
$$C' = \prod_{i=2}^{n_{\mathsf{rev}}}\delta_i'^{y_i} = \left(c_1{}^{-\langle \mathbf{x}_0,\mathbf{y}\rangle}\prod_{i=1}^{n_{\mathsf{rev}}} c_i^{y_i}\right)^{u_0}$$
and
$$\kappa_1' = \left(\frac{e(C',C_0')}{e(\zeta_0',C_0'')}\right)^{\frac{-1}{\langle \mathbf{x}_0,\mathbf{y}\rangle}} = e(g,g)^{\gamma_1 su_0}$$
similarly as described in the Correctness part .
2. It defines $I' \subset [l]$ such that $I' = \{i : \eta(i) \in S_0\}$, finds a set of constants $\{w_i' \in \mathbb{Z}_p\}_{i\in I'}$ satisfying $\sum_{i\in I'} w_i'\lambda_i = s$ where $\lambda_i = \langle N_i,\mathbf{v}\rangle$ are valid shares of the secret s according to $\Gamma = (N,\eta)$, N_i being the i-th row of the matrix N, s being the secret randomness picked to compute ct during encapsulation and $\mathbf{v} = (s,z_2,\ldots,z_d) \in \mathbb{Z}_p^d$. Since S_0 satisfies the access structure $\Gamma = (N,\eta)$, the set of constants $\{w_i' \in \mathbb{Z}_p\}_{i\in I}$ can be computed in polynomial time by Theorem 1.
3. Computes
$$\begin{aligned}
\kappa_2' &= \prod_{i\in I'}\left(e(C_i,\zeta_0)e(C_0',h_{\eta(i)}')\right)^{w_i'} \\
&= e(g,g)^{\gamma_0 a_0 s}
\end{aligned}$$
and recovers
$$\begin{aligned}
\frac{e(\zeta_1,C_0')}{\kappa_1'\kappa_2'} &= \frac{e(g^{\alpha_0}g^{\gamma_0 a_0}g^{\gamma_1 u_0}, g^s)}{e(g,g)^{\gamma_1 su_0}e(g,g)^{\gamma_0 a_0 s}} \\
&= e(g,g)^{\alpha_0 s}
\end{aligned}$$
in a similar way as described in the Correctness part. Here ζ_0,ζ_1 and $h_{\eta(i)}', i \in I'$ are extracted from $\mathsf{sk}_{\mathsf{id}_0,S_0}$ and $C_0', C_i, i \in I'$ are obtained from ct.

Key-homomorphism. Note that each of $(\mathcal{RS}',*)$, $(\mathcal{MSK},+)$, $(\mathcal{SK},\otimes)$ and $(\mathcal{EK},\odot)$ in our scheme RCP-ABKEM form a group where the operation $*$ on $\mathcal{RS}'$ is defined as $\tau * \tau_0 = (u+u_0, a+a_0)$ for $\tau = (u,a) \in \mathbb{Z}_p^2$ and $\tau_0 = (u_0,a_0) \in \mathbb{Z}_p^2$, the operation $\otimes$ on $\mathcal{SK} = \mathbb{G}^{n_{\mathsf{rev}}+|S|+2}$ is defined as coordinate-wise multiplication over $\mathbb{G}$, the operation $\odot$ on $\mathcal{EK} = \mathbb{G}_1$ is defined as multiplication over $\mathbb{G}_1$. The encapsulated key in our scheme is $k = f(\mathsf{msk},s) = e(g,g)^{\mathsf{msk}.s}$ where s is one of the randomness used in encapsulation algorithm. We also observe that for any identity $\mathsf{id} \in \mathbb{Z}_p$ with attribute set $S \subseteq \mathcal{AS}$, master secret keys $\mathsf{msk} = \alpha$, $\mathsf{msk}_0 = \alpha_0 \in \mathbb{Z}_p$ and randomness $\tau = (u,a) \in \mathbb{Z}_p^2$, $\tau_0 = (u_0,a_0) \in \mathbb{Z}_p^2$, $s \in \mathbb{Z}_p$, the following holds.

$$\mathsf{KeyExt}\Big(\mathsf{pp}, \mathsf{msk} = \alpha, \mathsf{id}, S; \tau = (u, a)\Big)$$
$$\otimes \mathsf{KeyExt}\Big(\mathsf{pp}, \mathsf{msk}_0 = \alpha_0, \mathsf{id}, S; \tau_0 = (u_0, a_0)\Big)$$
$$= \left(g^a, g^u, g^{\alpha} g^{\gamma_0 a} g^{\gamma_1 u}, \{p_x^a\}_{x\in S}, \{(c_1{}^{-\mathsf{id}^{i-1}} c_i)^u\}_{i=2}^{n_{\mathsf{rev}}}\right) \otimes$$
$$\left(g^{a_0}, g^{u_0}, g^{\alpha_0} g^{\gamma_0 a_0} g^{\gamma_1 u_0}, \{p_x^{a_0}\}_{x\in S}, \{(c_1{}^{-\mathsf{id}^{i-1}} c_i)^{u_0}\}_{i=2}^{n_{\mathsf{rev}}}\right)$$
$$= \Big(\; g^{a+a_0}, \;\; g^{u+u_0}, \;\; g^{\alpha+\alpha_0} g^{\gamma_0 (a+a_0)} g^{\gamma_1 (u+u_0)}, \;\; \{p_x^{a+a_0}\}_{x\in S},$$
$$\{(c_1{}^{-\mathsf{id}^{i-1}} c_i)^{u+u_0}\}_{i=2}^{n_{\mathsf{rev}}} \Big)$$
$$= \mathsf{KeyExt}\Big(\mathsf{pp}, \mathsf{msk} + \mathsf{msk}_0 = \alpha + \alpha_0, \mathsf{id}, S; \tau * \tau_0 = (u + u_0, a + a_0)\Big)$$

and

$$\begin{aligned} & f(\mathsf{msk} = \alpha, s) \odot f(\mathsf{msk}_0 = \alpha_0, s) \\ &= k \odot \widehat{k} \\ &= e(g,g)^{\alpha s} \odot e(g,g)^{\alpha_0 s} \\ &= e(g,g)^{(\alpha+\alpha_0)s} \\ &= f(\mathsf{msk} + \mathsf{msk}_0 = \alpha + \alpha_0, s) \end{aligned}$$

Besides, our scheme satisfies extended correctness as shown above. Hence, it achieves key-homomorphic property as defined in Section 2.4.

4 SECURITY ANALYSIS

Theorem 2. *Assuming that q-Decisional Bilinear Diffie-Hellman Exponent Problem* (DBDHE) *is hard, no probabilistic polynomial time adversary can selectively break the revocable ciphertext-policy attribute-based key encapsulation mechanism* (RCP-ABKEM) *decribed in section 3 with a challenge matrix of size $l^\star \times d^\star$ where $l^\star, d^\star \leq q$, a challenge revocation list $\mathcal{R}^\star$ where $|\mathcal{R}^\star| \leq q-2$.*

Proof. Due to page limit, the full proof will be appeared in the full version of this paper.

5 CONCLUSIONS

In this work, we have designed a refined version of RCP-ABKEM that satifies two additional properties *extended correctness* and *key-homomorphism*. We have provided an instantiation of this primitive from the hardness of q-DBDHE problem. Moreover, our scheme outperforms the existing similar schemes in terms of master secret key and ciphertext size.

REFERENCES

Attrapadung, N. and Imai, H. (2009). Conjunctive broadcast and attribute-based encryption. In *International conference on pairing-based cryptography*, pages 248–265. Springer.

Balu, A. and Kuppusamy, K. (2013). Ciphertext-policy attribute-based encryption with user revocation support. In *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pages 696–705. Springer.

Beimel, A. et al. (1996). Secure schemes for secret sharing and key distribution.

Boldyreva, A., Goyal, V., and Kumar, V. (2008). Identity-based encryption with efficient revocation. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 417–426.

Boneh, D. and Franklin, M. (2001). Identity-based encryption from the weil pairing. In *Annual international cryptology conference*, pages 213–229. Springer.

Liu, J. K., Yuen, T. H., Zhang, P., and Liang, K. (2018). Time-based direct revocable ciphertext-policy attribute-based encryption with short revocation list. In *International Conference on Applied Cryptography and Network Security*, pages 516–534. Springer.

Liu, Z., Wang, F., Chen, K., and Tang, F. (2020). A new user revocable ciphertext-policy attribute-based encryption with ciphertext update. *Security and Communication Networks*, 2020.

Liu, Z. and Wong, D. S. (2015). Practical ciphertext-policy attribute-based encryption: traitor tracing, revocation, and large universe. In *International Conference on Applied Cryptography and Network Security*, pages 127–146. Springer.

Sahai, A., Seyalioglu, H., and Waters, B. (2012). Dynamic credentials and ciphertext delegation for attribute-based encryption. In *Annual Cryptology Conference*, pages 199–217. Springer.

Sun, S., Sakzad, A., Steinfeld, R., Liu, J. K., and Gu, D. (2020). Public-key puncturable encryption: Modular and compact constructions. *Public Key Cryptography (1)*, 1210:309–308.

Wang, W., Wang, Z., Li, B., Dong, Q., and Huang, D. (2017). Ir-cp-abe: Identity revocable ciphertext-policy attribute-based encryption for flexible secure group-based communication. *IACR Cryptol. ePrint Arch.*, 2017:1100.