Key Policy-Attribute Based Fully Homomorphic Encryption (KP-ABFHE) Scheme for Securing Cloud Application in Multi-users Environment

Soo Fun Tan and Azman Samsudin

Abstract Recently, cloud technologies has become a cost-effective data solution among the small and medium-sized enterprises (SMEs). However, there is a raising concern on its security. This paper proposed the Key Policy-Attribute Based Fully Homomorphic Encryption (KP-ABFHE) scheme for providing an end-to end data protection in multi-users cloud environments. The proposed KP-ABFHE scheme is able to perform the computation while providing fine-grained access on the encrypted data. The proposed scheme is able to handle a monotonic access structure over a set of authorized attributes, without sacrificing the computation capabilities of homomorphic encryption. In addition, this paper proves that the proposed scheme is secure under a selective-set model with the hardness of Decision Ring-LWE$_{d,q,\chi}$ problem.

Keywords Cloud computing · Fully homomorphic encryption · Attribute-based encryption · Attribute-based fully homomorphic encryption

1 Introduction

Cloud computing has recently emerged as a cost-effective business data solutions among Small and Medium-sized Enterprises (SMEs). Cloud computing provides SMEs to store and process their data in the third-party data centers with a various service models, such as, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). While the cloud computing provides a numerous business competitive advantages, it raises a critical concern on data confidentiality issues, especially for SaaS model. SMEs are delegating their data access

S.F. Tan (✉) · A. Samsudin
School of Computer Sciences, Universiti Sains Malaysia, 11800 Penang, Malaysia
e-mail: soofuntan@gmail.com

A. Samsudin
e-mail: azman.samsudin@usm.my

H. Ibrahim et al. (eds.), *9th International Conference on Robotic, Vision, Signal Processing and Power Applications*, Lecture Notes in Electrical Engineering 398,
DOI 10.1007/978-981-10-1721-6_9

to a third-party Cloud Service Provider (CSP), who can abuse the access privilege in order to infer or sabotage valuable information. To protect these outsourced data from unauthorized access, one of the recent alternative is to exploit the usage of conventional security mechanisms. Tamper resistant hardware can be installed in an un-trusted third-party side or the private data can be encrypted before uploading onto CSPs.

However, complication arises when there is a need to do some computations on these confidential (encrypted) data such as predictive analysis, regression analysis and others. Both tamper-resistant hardware and conventional cryptosystems that provides data-at-rest protection by denying access on these encrypted data are not capable to support such computation. A trivial solution is to download the encrypted data, decrypt them before performing computation, which is impractical and problematic.

Apparently, homomorphic encryption, a scheme that is capable to compute over an encrypted data, is becoming an active research area to tackle this cloud security problem. To ensure the data confidentiality and privacy, SMEs encrypt the data with the homomorphic encryption scheme before uploading onto the CSP data centers. Meanwhile, the CSP is able to perform computation on these encrypted data without having to decrypt them.

Homomorphic encryption scheme was firs introduced by Rivest [1] in 1978. Over the last three decades, there is a very little progress in homomorphic encryption research until the Fully Homomorphic Encryption (FHE) was theoretically demonstrated by Gentry [2] in 2009. Existing works on homomorphic encryption scheme are directed towards the improvement of the speed performance, reducing the lengthy public key as well as the ciphertext size. However, most of the schemes are only capable of supporting a single owner content, in which all data are encrypted with a single user's secret key. In the real-world situation, SMEs' data are collected from various sources (e.g. different front-end devices, different locations, etc.) and accessible by many users with a different access rights. For supporting the multi-users environment, data owner has to share his secret key with other data users. However, sharing the same secret key among data users is an integrity threat, and should be avoided.

To bridge this gap, this paper proposes a Key Policy-Attribute Based Fully Homomorphic Encryption (KP-ABFHE) scheme. The KP-ABFHE scheme is a hybrid of the homomorphic encryption scheme and the attribute-based encryption scheme. Compared to conventional Access Control Systems (ACS) that requires high server-availability for granting access to the users, the Attribute-based Encryption (ABE) scheme is another cryptography advances that allows fine-grain access control on encrypted data with marginal operation cost on the server side.

This paper is organized as follows: Sect. 2 reviews the recent works on attribute-based homomorphic encryption scheme; Sect. 3 presents the preliminaries of the proposed scheme; Sect. 4 proposes KP-ABFHE scheme; Sect. 5 presents the security analysis of the proposed KP-ABFHE scheme and lastly ended with conclusion in Sect. 6.

2 Related Works

The extension to the homomorphic ecryption scheme that can support multi-users was first demonstrated by Xiao et al. [3] in 2012. In their symmetric homomorphic encryption scheme, the data collected from different sources are encrypted with the user's secret key, which is derived from a master key. Then, the user sends the encrypted data to key agent. The key agent then performs a secret-master key transformation by switching the encryption key from the user's secret key to the master key, without accessing the encrypted data. The key agent subsequently sends these newly re-encrypted data to CSP for storage and processing. Consequently, all data that stored on CSP data centers are encrypted under the same master key, and homomorphic evaluation can be conducted. Similarly, the computed result is forwarded to key agent for performing the master-secret key transformation before sending back the result to the user. Obviously, Xiao scheme [3] requires a high server availability, therefore resulting a performance bottleneck on their system, especially during the high network traffic period.

On the same year, another notable work was proposed by Lopez et al. [4]. Instead of working on symmetric approach, they proposed asymmetric homomorphic encryption scheme, which known as multi-key FHE. The main advantage of Lopez scheme [4] is that the scheme allows the CSP to perform the homomorphic evaluation on the data that was encrypted with different user's secret keys. Similar to Xiao et al. scheme [3], the collected data from different sources are encrypted with the user's secret keys. However, instead of having a key agent act as a mediator to perform the key transformation, Lopez scheme [4] delegates this task to the CSP server. When there is a user request to perform jointly-computation on the encrypted data, the CSP first switches the user's secret key to an evaluation key by using the relinearization technique that was introduced earlier by Brakerski and Vaikuntanathan [5]. Subsequently, CSP is able to perform homomorphic evaluation since all data are now encrypted under the same evaluation key. The computed ciphertext can be jointly decrypted using the secret key of all users that involved in the computation. The Lopez scheme [4] eliminates the bottleneck of key agent as in Xiao scheme [3]; however, introduces another performance overhead on both user and CSP server. Each request of computation involves the evaluation key generation and re-encryption of the data on user and CSP server respectively.

At CRYPTO 2013, Gentry et al. [6] proposed the homomorphic encryption with a more sophisticated access control capabilities. Compared to previous scheme [3, 4] that works on multi-key level, Gentry scheme [6] proposed an Identity-Based Fully Homomorphic Encryption (IBFHE) scheme by integrating the homomorphic encryption scheme with the Identity-Based Encryption (IBE). In addition, Gentry scheme [6] can be extended to provide a fine-grained access control on encrypted data, which was encrypted based on the user's attributes. However, their homomorphic evaluation can only support data that was previously encrypted under the same attributes.

A fine-grain access control on encrypted data was further demonstrated by Clear and McGoldrick [7, 8]. They hybrided the homomorphic encryption scheme with the Ciphertext-Policy ABE scheme [7]. However, the homomorphic evaluation is limited to only a single attribute encrypted data. For supporting computation on the data that with multi-attributes, Clear and McGoldrick [8] subsequently borrow the key transformation concept from Lopez scheme [4] to switch the "attribute" during homomorphic evaluation.

In order to support multi-users requirement, all the schemes reviewed so far suffer from the fact that the introduced extension to the homomorphic encryption scheme had a negative effect on the computation capabilities. Both Xiao scheme [3] and Lopez scheme [4] require an additional key transformation process either by the key agent [3] or by the CSP server [4]. Meanwhile, Gentry scheme [6] and Clear-McGoldrick scheme [7] are able to provide a fine-grain control access on the encrypted with user's attributes. However, both Gentry scheme [6] and Clear-McGoldrick scheme [7] only allow the computation to be performed on the data encrypted with the same attribute, thus unable to support homomorphic encryption under multi-users setting. Although the Clear-McGoldrick scheme [8] is capable to support multi-attribute computation, however, their scheme inherits the performance bottleneck from the Lopez scheme [4].

To solve this problem, instead of fusing both homomorphic encryption and ABE directly into a single ciphertext as implemented in the previous works; this paper proposes the KP-ABFHE scheme in which a ciphertext is made out of two sub-components. The first component carries the data, encrypted with a homomorphic encryption scheme; while the second component enable a fine-grains access on the encrypted data with ABE scheme. Thus, enables the extension to support multi-users environment without affecting the computation capabilities of the homomorphic encryption.

3 Preliminaries

This section introduces some concepts and background which will be used in the construction of the proposed KP-ABFHE scheme (see Sect. 4).

Definition 1 (*Monotonic Access Structure* [9]) Let $U = \{u_1, u_2, \ldots, u_n\}$ be a set of attributes. A collection $\mathbb{A} \subseteq 2^U$ is monotone if $\forall B, C : B \in \mathbb{A}$ and $B \subseteq C$ implies $C \in \mathbb{A}$. A monotone access structure is a monotone collection $\mathbb{A}$ of non-empty subsets of $\{u_1, u_2, \ldots, u_n\}$. The sets in $\mathbb{A}$ are called as authorized sets, and the sets not in $\mathbb{A}$ are called as unauthorized sets.

The access structure, $\mathbb{A}$ can be realized with a secret sharing scheme where each authorized attribute holds a private pieces of secret key, SK. Any authorized set of attribute can reconstruct the secret from its pieces, and any unauthorized set not in $\mathbb{A}$ cannot reveal any partial information about the secret. For handling a monotone

access structure, the Linear Secret Sharing Scheme (LSSS) defined as follows, will be used in the proposed scheme. It has been proven that any monotone access structure can be realized with LSSS in [9].

Definition 2 (*Linear Secret Scheme (LSSS)* [9]) A secret sharing scheme $\prod$ over a set of attributes $U = \{u_1, u_2, \ldots, u_n\}$ is called linear (over R_q) if satisfying the following properties:

- The secret shares for each attribute form a vector over R_q;
- There exists a share-generating matrix for $\prod$, which denoted as Matrix $G \in R_q^{n \times m}$, with row labels $p(i) \in U$, $\forall i \in [n]$. Given a column vector, $\boldsymbol{v} = (SK, r_2, \ldots, r_m)$, where $SK \in R_q$ is the secret to be shared and $r_2, \ldots, r_m \leftarrow R_q$ randomly chosen, the Gv is the vector of n shares of the secret according to $\prod$. The share $\delta_i = (Gv)_i$, i.e., the inner product $G_i \cdot v$ belongs to attribute $p(i)$, where p is a function from $\{1, \ldots, n\}$ to U.

The LSSS enjoys the linear reconstruction property [9] as follows. Suppose that $\prod$ is a LSSS that represents the access structure $\mathbf{A}$. Let $\mathbb{A} \in \mathbf{A}$ be an authorized set, and $I \subset \{1, 2, \ldots, n\}$ be defined as $I = \{i : p(i) \in \mathbb{A}\}$. There exist constants $\{w_i \in R_q\}_{i \in I}$ such that of δ_i are valid shares of a secret, SK according to $\prod$, then $\sum_{i \in I} \delta_i w_i$. Furthermore, these constants w_i can be found in polynomial time in the size of share-generating matrix G. For any unauthorized set, no such constants exists. In this paper, the LSSS matrix (G, p) will be used to express an access structure associated to the user's secret key.

4 Construction of the Key Policy-Attribute Based Homomorphic Encryption (KP-ABFHE) Scheme

The homomorphic encryption scheme is constructed based on Ring-LWE problem [10, 11]; whereas, the ABE scheme is an extension of [12]. The main differences between the proposed scheme with the existing works [3, 4, 6–8] are two fold. First, compared to existing ABHE schemes [6–8] that only considered their access structure as a single attribute [6, 7] ; or use a single attribute to represent a set of "sub-attributes" [8]; the proposed scheme uses the LSSS matrix (G, p) to express a monotonic access structure over a set of attributes. Second, instead of fusing both homomorphic encryption and ABE scheme directly into a single ciphertext, the proposed KP-ABFHE scheme decomposes a ciphertext into two sub-components. The first component encrypts the data with a public key, PK; whereas, the second component consists a set of authorized attributes, $\mathbf{A}$. The homomorphic evaluation only involves the first component. Meantime, the second component is used to fine-grained an access control on the encrypted data. The user's secret key, SK is associated with a monotonic access structure, $\mathbb{A}$ over a set of authorized attributes. Each authorized attribute in the monotonic access structure, $\mathbb{A}$ holds a valid shares of the secret key, SK. Thus, the encrypted data can be decrypted correctly if and only if

$\mathbf{A} \in \mathbb{A}$. In addition, for preventing the collusion attacks among multiple users, the proposed KP-ABFHE scheme extended the secret key randomization techniques [13, 14] to blind the user's secret key, $SK_{\mathbb{A}}$, in which the multiple users are unable to pool their attributes together and re-construct a valid secret key, $SK_{\mathbb{A}}$. Next, the proposed KP-ABFHE scheme is formally defined as follows.

***Setup*(λ, *U*, $\mathcal{K}$) → (*PP*, *MSK*)**. The setup algorithm takes as input: a security parameter, λ, a universe of attributes, $U = \{u_1, u_2, \ldots, u_n\}$ and the number of users, $\mathcal{K}$ in the systems. Choose a sufficiently large prime modulus $q = 1 \bmod (2\lambda)$, and a smaller positive integer p, such that $p \ll q$ and $\gcd(p, q) = 1$. Let $f(x) = (x^d + 1)$, where d is a power of 2. Let $\mathbb{R}_q = \mathbb{Z}_q[x] / < f(x) >$ be the ring of integer polynomials modulo both $f(x)$ and q. Let $\chi = \chi(\lambda)$ be an error distribution over $\mathbb{R}_q$. Select a uniformly random master secret key, $SK_0 \leftarrow \mathbb{R}_q$ and a number t random elements, $a_t \leftarrow \mathbb{R}_q$ and error terms, $e_t \leftarrow \chi$. Compute a number of t public keys, such that $PK_t = a_t \cdots SK + pe_t \in \mathbb{R}_q$. Next, for each attribute $\{u_1, u_2, \ldots, u_n\}$ in U, select a pair of uniformly random $(K_i, K_i^{-1}) \leftarrow \mathbb{R}_q$, where K_i^{-1} is the inverse of K_i in $\mathbb{R}_q$ and a small error term, $g_i \leftarrow \chi$. Compute $PK_i = K_i + pg_i \in R_q$. Lastly, outputs the public parameters PP, and a master secret key, MSK as follows.

$$PP = \{a_t, PK_t, \{PK_i\}_{i=1}^n\}$$
$$MSK = \{SK_0, \{K_i\}_{i=1}^n, \{K_i^{-1}\}_{i=1}^n\}$$

***KeyGen*(*PP*, *MSK*, $\mathbb{A}$) → ($SK_{\mathbb{A}}$)**. The key generation algorithm takes the public parameters, PP, a master secret key, MSK and an access structure, $\mathbb{A}$ over the universe of attributes U as input. It first transforms $\mathbb{A}$ into the LSSS matrix (G, p), where matrix $G \in R_q^{n \times m}$, with row labels $p(i) \in U, \forall i \in [n]$. Then, it distributes the valid shares of master secret key, SK by generating a vector, $\boldsymbol{v}$ such that $\boldsymbol{v} = (SK, r_2, \ldots, r_m)$, where $r_2, \ldots, r_m \leftarrow \mathbb{R}_q$ is randomly chosen. The Gv is the vector of n shares of the secret key, SK according to secret sharing scheme $\prod$ over $\mathbb{A}$. For each $i = 1$ to n, calculates the secret share, $\delta_i = G_i \times v \in \mathbb{R}_q$, where Gi is the vector corresponding to i-th row of G. Next, select a uniformly random, α and its inverse, α^{-1} such that $\alpha, \alpha^{-1} \leftarrow \mathbb{R}_q$ and error terms, $h_t, j_i \leftarrow \chi$. The algorithm outputs the user's secret key, $SK_{\mathbb{A}} = (SK_0, SK_i)$ associated with a description of (G, p), where:

$$SK_0 = \alpha^{-1} + ph_t \in \mathbb{R}_q$$
$$SK_i = \alpha \cdot K_i^{-1} \cdot \delta_i + pj_i \in \mathbb{R}_q, \forall i \in \mathbb{A}$$

***Encrypt*($M_1, \ldots, M_t$, *MSK*, $\mathbf{A}$) → ($CT_1, \ldots, CT_t$)**. The encryption algorithm takes the master secret key MSK, the messages, $M \in \{1, 0\}$, and a set of authorized attributes, $\mathbb{A}$. Next, select a uniformly random, $r_t \leftarrow R_q$ and error terms, $x_t, y_i \leftarrow \chi$. It outputs the ciphertexts, $CT_t = (C_t^0, C_t^i)$, where:

$$C_t^0 = PK_t \cdot r_t + M_t + px_t \in R_q$$
$$C_t^i = a_t \cdot PK_i \cdot r_t + py_i \in R_q$$

***Evaluate*(*PP, F*, $C_1^0, \ldots, C_t^0$) → $F(C_1^0, \ldots, C_t^0)$**. The evaluation algorithm takes a public parameter, *PP*, a polynomial time computation function, *F* and the first component of ciphertexts, $C_1^0, \ldots, C_t^0$, as input. It outputs the computed result in ciphertext space, such that $C_t^{0*} = F(C_1^0, \ldots, C_t^0)$.

***Decrypt*($SK_{\mathbb{A}}, C_t^{0*}, C_t^i$) → ($M^*$ *or* ⊥)**. The decryption algorithm takes a computed result in ciphertext space, $C_t^{0*} = F(C_1^0, \ldots, C_t^0)$ and a user's private key, $SK_{\mathbb{A}}$ as input. It recovers the computed result in plaintext space, M^*, such that $C_t^{0*} = F(C_1^0, \ldots, C_t^0) = F(M_1, \ldots, M_t)$. Compute a set of constants, $\{w_i \in R_q\}_{i \in I}$ with a linear reconstruction algorithm of LSSS such that if $\{\delta_i\}$ are valid shares of shared secret, *SK* according to *G*, then $\sum_{i \in I} \delta_i w_i = SK$. Next, compute $M^{*\prime} = C_t^{0*} - SK_0 \sum_{i \in I} SK_i \cdot w_i \cdot C_t^i$ and outputs the computed result in plaintext space, such that $M^* = M^{*\prime} \bmod p$, otherwise, outputs a false symbol, ⊥.

Correctness. The correctness of the proposed KP-ABFHE scheme follows the correctness of the LSSS linear reconstruction property [9]. If $\mathbf{A} \in \mathbb{A}$ be an authorized set, and $I \subset \{1, 2, \ldots, n\}$ be defined as $I = \{i : p(i) \in \mathbb{A}\}$, then there exists a set of constants $\{w_i \in R_q\}_{i \in I}$ and $\sum_{i \in I} \delta_i w_i = SK$. Similar to most of the lattice based encryption schemes, the encryption algorithm of the proposed KP-ABFHE scheme involves adding the error terms into ciphertexts. Therefore, for ensuring the correctness of the *Decrypt* algorithm, the error terms $(e_t, g_i, h_t, j_i, x_t, y_i)$ of the ciphertext must be small enough compared to the ratio *q* to *p*, denote as $\Delta = \lfloor q/p \rfloor$.

Computation on Encrypted Data. The computation on encrypted data is implemented with evaluation algorithm, $Evaluate(PP, F, C_1^0, \ldots, C_t^0)$. The homomorphic addition properties, f_{add}, exists as the following.

$$\begin{aligned} f_{add}(C_1^0, C_2^0) &= C_1^0 + C_2^0 \\ &= (PK_1 \cdot r_1 + M_1 + px_1) + (PK_2 \cdot r_2 + M_2 + px_2) \\ &= M_1 + M_2 + PK_1 r_1 + PK_2 r_2 + p(x_1 + x_2) \end{aligned}$$

Meanwhile, the homomorphic multiplication properties, f_{mult} can be founded as follows.

$$\begin{aligned} f_{mult}(C_1^0, C_2^0) &= C_1^0 \times C_2^0 \\ &= (PK_1 \cdot r_1 + M_1 + px_1) \times (PK_2 \cdot r_2 + M_2 + px_2) \\ &= M_1 M_2 + PK_1 r_1 (PK_2 r_2 + M_2 + px_2) + PK_2 r_2 (M_1 + px_1) \\ &\quad + px_1 (M_2 + px_2) + px_2 M_1 \end{aligned}$$

The computed result in both homomorphic addition, $f_{add}(C_1^0, C_2^0)$ and homomorphic multiplication $f_{mult}(C_1^0, C_2^0)$, can be recovered with the decryption algorithm, *Decrypt*. As aforementioned, the homomorphic evaluation of the proposed KP-ABFHE scheme only involves the first component of ciphertext, C_t^0. Thus, the proposed KP-ABFHE scheme allows the homomorphic evaluation can be conducted as in the origin scheme [10, 11], without affecting its computation capabilities.

Fine-Grained Access Control on Encrypted Data. The access control of the encrypted data is implemented by the second component of ciphertext, C_t^i. Any superset of the set $\mathbf{A}$ of encrypted data that satisfying the access structure, $\mathbb{A}$ of the user's secret key, $SK_{\mathbb{A}}$ is able to recover the computed result. If $\mathbf{A}$ is a set of attributes satisfying an access structure $\mathbb{A}$, then B such that $A \subseteq B$ also satisfies $\mathbb{A}$. For instance, consider a simple monotonic access structure $\mathbb{A} = \{u_1 \cap u_2\}$, if $\mathbf{A} = \{u_1, u_2\}$ satisfying an access structure $\mathbb{A}$, then $\mathbf{B} = \{u_1, u_2, u_3\}$ also satisfying $\mathbb{A}$.

5 Security Analysis

The security of the KP-ABFHE scheme is constructed based on the hardness of Ring-LWE problem. This section proves that the KP-ABFHE scheme is secure under a selective-set model with the hardness of Decision Ring-LWE$_{d,q,\chi}$ problem as follows.

Theorem 1 *If there exists an adversary Probabilistic Polynomial Time (PPT) algorithm, $\mathcal{A}$ with the advantage, ϵ in selective-set model for the KP-ABFHE scheme as aforementioned construction in Sect. 3, then there exists a PPT algorithm simulator, $\mathcal{B}$ that decides the Decision Ring-LWE$_{d,q,\chi}$ problem with advantage $\frac{\epsilon}{2}$.*

Proof As described in [10, 11], the Decision Ring-LWE$_{d,q,\chi}$ problem instance is conditioned as sample oracle $\mathcal{O}$, that can be either a noisy pseudo-random, $\mathcal{O}_s$ for master secret key $SK \leftarrow \mathbb{R}_q$, or truly random $\mathcal{O}_\$$. The Decision Ring-LWE$_{d,q,\chi}$ problem allows repeated queries to be sent to the challenge oracle. The adversary algorithm $\mathcal{A}$ decides the Decision Ring-LWE$_{d,q,\chi}$ problem if $|Pr[\mathcal{A}^{\mathcal{O}_s} = 1] - Pr[\mathcal{A}^{\mathcal{O}_\$} = 1]|$ is non-negligible for a random $SK \in R_q$.

Meanwhile, the advantage of $\mathcal{A}$ in selective-set model [13, 15] is defined as $adv(\mathcal{A}) = |Pr[r' = r] - \frac{1}{2}|$. When the Ring-LWE$_{d,q,\chi}$ oracle $\mathcal{O}$ is a noisy pseudo-random $\mathcal{O}_s$, the $\mathcal{A}$ has an advantage ϵ, then $|Pr[r' = r|\mathcal{O} = \mathcal{O}_s]| = \frac{1}{2} + \epsilon$ and $Pr[\mathcal{O}' = \mathcal{O}|\mathcal{O} = \mathcal{O}_s] = \frac{1}{2} + \epsilon$. Whereas, Ring-LWE$_{d,q,\chi}$ oracle $\mathcal{O}$ is a truly random $\mathcal{O}_\$$, the $\mathcal{A}$ has no advantage ϵ and has no any idea regarding the r, then $Pr[r' \neq r|\mathcal{O} = \mathcal{O}_\$] = \frac{1}{2}$ and $Pr[\mathcal{O}' = \mathcal{O}|\mathcal{O} = \mathcal{O}_\$] = \frac{1}{2}$.

The advantage ϵ of simulator, $\mathcal{B}$ in this selective game model under the Decision Ring-LWE$_{d,q,\chi}$ problem thereby is as follows.

$$\begin{aligned} &\tfrac{1}{2}Pr[\mathcal{O}' = \mathcal{O}|\mathcal{O} = \mathcal{O}_s] + \tfrac{1}{2}Pr[\mathcal{O}' = \mathcal{O}|\mathcal{O} = \mathcal{O}_\$] - \tfrac{1}{2} \\ &= \tfrac{1}{2}(\tfrac{1}{2} + \epsilon) + \tfrac{1}{2}(\tfrac{1}{2}) - \tfrac{1}{2} \\ &= \tfrac{\epsilon}{2} \end{aligned}$$

Hence, this concludes the proof of security reduction that there exists a PPT algorithm simulator, $\mathcal{B}$ that decides the Decision Ring-LWE$_{d,q,\chi}$ problem with advantage $\frac{\epsilon}{2}$.

6 Conclusion

This paper presents the construction of the proposed KP-ABFHE scheme based on the hardness of Decision Ring-LWE$_{d,q,\chi}$ problem. For supporting end-to end data protection in multi-users cloud environments, this paper extended the homomorphic encryption scheme with the attribute-based encryption scheme. The proposed KP-ABFHE scheme is able to perform the computation while providing fine-grained access control on the encrypted data. In terms of access structure, the proposed KP-ABFHE scheme is much flexible in which it is able to handle a monotonic access structure over a set of authorized attributes. In terms of encryption, the proposed KP-ABFHE scheme decomposes a ciphertext into two sub-components, which in turn improved the computation capabilities of the homomorphic encryption. The first component carries the data encrypted with homomorphic encryption scheme; whereas, the second component is responsible to provide a fine-grained access on encrypted data by using the attribute-based encryption scheme. Thus, enables the extension of homomorphic encryption scheme to support multi-users environment without affecting the computation capabilities of homomorphic encryption. The proposed scheme is secure under a selective-set model with the hardness of Decision Ring-LWE$_{d,q,\chi}$. For future works, several aspect of this work can be further explored, such as the implementation of fully hommorphic encryption with various noise management techniques: boostrapping, modulus switching or flattening techniques.

References

1. Rivest R (1978) On data banks and privacy homomorphisms. Found Sec Comput 4:169–180
2. Gentry C (2009) A Fully Homomorphic encryption scheme. Phd Thesis. Standford University
3. Xiao L, Bastani O, Yen I (2012) An efficient homomorphic encryption protocol for multi-user systems. IACR Crypt. ePrint Arch
4. Lopez-Alt A, Tromer E, Vaikuntanathan V (2012) On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In: 44th symposium on Theory of Computing. ACM press, New York, pp 12–19
5. Brakerski Z, Vaikuntanathan V (2011) Fully homomorphic encryption from ring-lwe and security for key dependent messages. In: Rogaway P (ed) CRYPTO 2011, vol 6841, LNCS. Springer, Heidelberg, pp 505–524
6. Gentry C, Sahai A, Waters B (2013) Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Ca-netti R, Garay JA (eds) CRYPTO 2013, vol 8042, LNCS. Springer, Heidelberg, pp 75–92
7. Clear M, McGoldrick C (2013) Policy-based non-interactive outsourcing of com-putation using multikey FHE and CP-ABE. In: 10th international conference on security and cryptography (SECRYPT). IEEE Press, Reykjavik, pp 444–452

8. Clear M, McGoldrick C (2013) Bootstrappable identity-based fully homomorphic encryption. In: Gritzalis D, Kiayias A, Askoxylakis I (eds) CANS 2013, vol 8831, LNCS. Springer, Switzerland, pp 1–19
9. Beimel A (1996) Secure schemes for secret sharing and key distribution. Phd Thesis. Israel Institute of Technology, Haifa, Israel
10. Lyubashevsky V, Peikert C, Regev O (2010) On ideal lattices and learning with errors over rings. In: Gilbert H (ed) EUROCRYPT 2010, vol 6110, LNCS. Springer, Heidelberg, pp 1–23
11. Lyubashevsky V, Peikert C, Regev O (2013) A toolkit for ring-LWE cryptography. In: JohanssonT, Nguyen PQ (eds) EUROCRYPT 2013, vol. 7881, LNCS. Springer, Heidelberg, pp 35–54
12. Soo Fun T, Azman S (2015) Lattice ciphertext-policy attribute-based encryption from ring-LWE. In: 2nd international symposium on technology management and emerging technologies (ISTMET). IEEE Press, Langkawi, pp 282–286
13. Goyal V, Pandey O, Sahai A, Waters B (2006) Attribute-based encryption for fine-grained access control of encrypted data. In: 13th ACM conference on Computer and communications security (CCS). ACM Press. New York, pp 89–98
14. Koo D, Hur J, Yoon H (2013) Secure and efficient data retrieval over encrypted data using attribute-based encryption in cloud storage. Comp. Elect. Eng. 39:34–46
15. Weiling Z, Jianping Y, Ting W, Peng Z, Weixin X (2014) Efficient attribute-based encryption from R-LWE. Chin. J. Elect. 23:4