JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 33, 675-694 (2017)
DOI: 10.6688/JISE.2017.33.3.5

Ciphertext Policy-Attribute Based Homomorphic Encryption (CP-ABHE$_{R\text{-}LWE}$) Scheme: A Fine-Grained Access Control on Outsourced Cloud Data Computation*

SOO-FUN TAN AND AZMAN SAMSUDIN
School of Computer Sciences
Universiti Sains Malaysia
Penang, 11800 Malaysia
E-mail: soofuntan@gmail.com; azman.samsudin@usm.my

Recently, homomorphic encryption is becoming one of the promising tools to protect outsourced data on cloud service providers. However, most of the existing homomorphic encryption schemes are designed to achieve Fully Homomorphic Encryption that aimed to support arbitrary computations for only single-data ownership scenario. To bridge these gaps, this paper proposed a non-circuit based Ciphertext Policy-Attribute Based Homomorphic Encryption (CP-ABHE$_{R\text{-}LWE}$) scheme to support outsourced cloud data computations with a fine-grained access control under the multi-user scenario. First, this paper incorporates Attribute Based Encryption (ABE) scheme into homomorphic encryption scheme in order to provide a fine-grained access control on encrypted data computation and storage. Then, the proposed CP-ABHE$_{R\text{-}LWE}$ scheme is further extended into non-circuit based approach in order to increase the practical efficiency between enterprise and cloud service providers. The result shows that the non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme has greatly reduced the computation time and ciphertext size as compared to circuit based approach. Subsequently, the proposed CP-ABHE$_{R\text{-}LWE}$ scheme was proven secure under a selective-set model with the hardness of Decision R-LWE$_{d,q,\chi}$ Problem.

Keywords: cloud security, data centric security, homomorphic encryption, attribute-based encryption, attribute-based homomorphic encryption (ABHE), lattice cryptography, ring-learning with errors (R-LWE or Ring-LWE)

1. INTRODUCTION

The recent advancement on cloud technologies promises a cost-effective, scalable and maintenance-free data solution for enterprises. However, moving sensitive data, as well as outsourcing critical analytical works to cloud service providers in turn exposes a number of security and privacy issues. To overcome these problems, the homomorphic encryption scheme with a mechanism similar to the conventional cryptography, but with added capabilities of computing over encrypted data, has become an active research area to secure these outsourced data storage and computation recently.

Started from Gentry [1] earlier work on Fully Homomorphic Encryption (FHE) in 2009, a number of homomorphic encryption schemes have been proposed. These schemes either: (i) inherit Gentry's blueprint [1] that focus on solving the overhead of bootstrapping algorithm [2-4]; (ii) simplify Gentry's work [1] into a modular arithmetic – DGHV

Received June 30, 2016; accepted October 10, 2016.
Communicated by Balamurugan Balusamy.
* This work was supported by a research grant from Universiti Sains Malaysia (USM) [1001/PKOMP/811334].

scheme [5-11]; (iii) incorporate Learning with Error (LWE) approach [37, 45, 51, 52]; or, iv) extend into more efficient ideal lattice approach (also called as R-LWE approach) [13-18, 52]. Whether these FHE schemes can be further adapted to secure a real-world outsourced data computation on the cloud service providers is still questionable, due to the following reasons.

First, almost all of these homomorphic encryption schemes [1-14, 16-19, 23-25, 37, 45, 51-53] are constructed based on circuit-based approach, in which every single bit of the plaintext is encrypted as a separate ciphertext (also known as bit-by-bit encryption [20, 21]). The main advantage of circuit-based approach is that all operations on various operands can be computed by constructing the corresponding circuits [37, 38], thus makes the realization of FHE to support arbitrary computation possible. However, these proposed works are still not efficient to support practical deployments, due to their high computation complexity, huge generated ciphertext and public key size [22-24]. Thus, increases the communication costs and bandwidths required when the outsourced data are transferred, back and forth, from the user's side to cloud service provider's side [23]. For instance, Gentry's scheme [4, 25] took more than 15 minutes to add two 32-bit integers and more than 18.6 hours to multiply them. The encryption of 4MB data generates a ciphertext with a size of more than 74TB [6] and even with the parallelism and compression technique [7] being used, the ciphertext size can only be further reduced to 280GB [54].

Second, majority of the existing homomorphic encryption schemes [1-14, 16-20, 22-25] only support a single-data owner scenario. All data are encrypted under the same secret key and accessed by a single user. In the real world of cloud services scenario, enterprises' data are collected from various sources (*e.g.* sensor network, social media, traditional databases, *etc.*) and are being accessed by multiple users either within the organization (*e.g.* manager, internal staff, *etc.*) or outside the organization (*e.g.* research partners, marketing affiliation, *etc.*). The key question here is how the homomorphic encryption scheme can be further extended to provide a fine-grained access control on these encrypted data. At the CRYPTO 2013 conference, Gentry *et al.* [26] presented the first Attribute-Based Fully Homomorphic Encryption (ABFHE) that extends Identity-Based FHE (IBFHE) to achieve the desired control. In the same year, Clear and McGoldrick [27] proposed another ABFHE scheme by incorporating properties from FHE and CP-ABE, and named it as Policy-Based Homomorphic Encryption (PBHE) scheme. Thereafter, Clear and McGoldrick [28] further extended their ABFHE scheme to support a multi-attribute computation scenarios. In general, the aim of these schemes [26-28] is to associate the homomorphic encryption scheme with attribute-based access control policy either by associating the secret key with the access policy [26, 28] (so called as Key-Based ABFHE, KP-ABFHE) or by associating the ciphertext with the access policy [27] (so called as Ciphertext-Based ABFHE, CP-ABFHE). However, unlike the original ABE scheme that is capable of handling a more expressive access structure such as AND gate, OR gate, *etc.*, these pilot ABHE schemes is only able to manipulate a single attribute rather than a set of attributes [27], which resulted in the need to encrypt every single bit of the plaintext with an attribute [26, 28]. Thus, limits their capabilities to have a fine-grained access control on these private data.

To bridge these gaps, this paper proposed a non-circuit based Ciphertext Policy-Attribute Based Homomorphic Encryption (CP-ABHE$_{\text{R-LWE}}$) scheme for securing the

outsourced cloud data solution with a fine-grained access control under the multi-user scenario. The contribution of this paper is therefore two-fold; (i) ***CP-ABFHE$_{R\text{-}LWE}$ scheme with monotonic access structure***. Incorporating the homomorphic encryption with ABE scheme always affecting the intricate part of the homomorphic computation. This is one of the main reasons why existing ABHE schemes use only a single attribute. To the best of our knowledge, there are no known ABHE scheme in the literature that are capable to efficiently handle a monotonic access structure − without affecting the capability of the homomorphic computation. To solve this tricky problem, the proposed scheme encrypts a message into two separate ciphertext components (C^0, C^1). The ciphertext C^0 is encoded with message, whereas ciphertext C^1 is associated with a monotonic access structure over a set of authorized attributes. The homomorphic computation can be conducted as usual since it only involves the ciphertext C^0. To decrypt, only an eligible user with the corresponding set of authorized attributes that satisfy the corresponding access structure can recover the secret key from ciphertext C^1 and retrieve the message correctly; (ii) ***Non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme***. In this paper, the proposed circuit based CP-ABHE$_{R\text{-}LWE}$ scheme is further extended onto a non-circuit based approach in order to support real-world cloud data processing environments. In the proposed scheme, the whole plaintext is encrypted as a single ciphertext, thus greatly reduce the computation time, ciphertext size, communication cost and bandwidth required to transfer them to cloud service providers.

The rest of this paper is organized as follows. Section 2 states the notations and background definitions used in this paper. Section 3 describes the proposed non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme. In Section 4, the security of the proposed scheme is analysed under the selective game model. Section 5 compares the performance of the proposed scheme with existing ABFHE schemes. Lastly, Section 6 conclusions.

2. PRELIMINARIES

This section introduces some concepts and definitions which will be used in the construction of the proposed CP-ABHE$_{R\text{-}LWE}$ scheme in Section 3.

2.1 Access Control

Definition 1 (Attribute [29]): An attribute is any distinctive feature, characteristic, or property of an object that can be identified or isolated quantitatively or qualitatively by either human or automated means. For instance, an attribute could be a gender (male or female), job position (physician, researcher, administrator, *etc.*) department (cardiology, gynecology, dermatology, *etc.*), *etc.*

Definition 2 (Access Structure [30]): Let $U = \{u_1, u_2, ..., u_n\}$ 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 $\{u_1, u_2, ..., u_n\}$. 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 piece of the secret key, s. Any authorized set of attributes 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 in the following will be used in the proposed scheme. It has been proven that any monotone access structure can be realized with LSSS [30].

Definition 3 (Linear Secret Scheme (LSSS) [30]): A secret sharing scheme $\prod$ over a set of attribute $U = \{u_1, u_2, ..., u_n\}$ is called linear (over R_q) if it satisfies the following properties:

- The shares for each attribute form a vector over R_q.
- There exists a share-generating matrix for $\prod$, which we 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, $v = (s, r_2, ..., r_m)$, where $s \in R_q$ is the secret to be shared and $r_2, ..., r_m \leftarrow R_q$ randomly chosen, the Gv is the vector of n shares of the secret according to $\prod\prod$. 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 $\{1, .., n\}$ to U.

The LSSS enjoys the linear reconstruction property [30] as follows. Suppose that $\prod$ is an LSSS that represents the access structure $\mathcal{A}$. Let $A \in \mathcal{A}$ be an authorized set, and I_c $\{1, 2, ..., n\}$ is defined as $I = \{i: p(i) \in A\}$. There exist constants $\{w_i \in R_q\}_{i \in I}$ such that for $\{\delta_i\}$ valid shares of a secret s, according to $\prod$, $\Sigma_{i \in I} \delta_i w_i$. 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 exist. In this paper, the LSSS matrix (G, p) will be used to express an access structure associated to a ciphertext.

2.2 Attributed Based Encryption (ABE) Scheme

In general, Attributed Based Encryption (ABE) scheme provides a fine-grained control access on encrypted data by associating the access structure either with the user's secret key (so called as Key Policy-ABE, KP-ABE) or ciphertext (so called as Ciphertext Policy-ABE, CP-ABE) [31]. Recently, CP-ABE is increasingly becoming dominant in the literature due to its similarity to the real-world access control (*e.g.* role-based access control) [32, 33]. Furthermore, in CP-ABE scheme, data can be encrypted before knowing the user authorization, as long as the access policy is available. Thus, CP-ABE scheme is better in supporting scalability for the cloud processing. This paper employs CP-ABE to realize implicit authorization on encrypted data. CP-ABE is defined as follows.

Definition 4 (Ciphertext Policy Attribute-based Encryption (CP-ABE) [34]): A Ciphertext Policy Attribute-Based Encryption scheme consists of four algorithms as follows:

Setup (λ, U) The setup algorithm takes security parameter λ, and a universe of attributes U as input. It produces the public parameters PP, and a master key MK.

Encrypt (PP, M, $\mathcal{A}$) The encryption algorithm takes the public parameters PP, a message M, and an access structure $\mathcal{A}$, over the universe of attributes U, as input. The algorithm encrypts M and outputs a ciphertext CT such that only the eligible user who is able

to produce the set of attributes that satisfies the access structure will be able to decrypt the message.

KeyGen (MK, A) The key generation algorithm takes the master key MK, and a set of attributes A, that describes the key. It produces a private key SK for A.

Decrypt (PP, CT, SK) The decryption algorithm takes the public parameters PP, a ciphertext CT that consists of the access structure $\mathbb{A}$, and a private key SK of attributes set, A. If the attributes set, A, satisfies the access structure $\mathbb{A}$ such that $A \in \mathbb{A}$, then the algorithm will produce the message M, otherwise, a false symbol $\perp$, will be generated.

2.3 Homomorphic Encryption (HE) Scheme

Homomorphic encryption scheme allows computation to be performed on ciphertext without the need for the ciphertext to be decrypted, it is formally defined as follows.

Definition 5 (Homomorphic Encryption): A homomorphic encryption scheme consists of five algorithms (*Setup*, *KeyGen*, *Encrypt*, *Evaluate*, *Decrypt*). They are as follows.

KeyGen (1^{λ}) The key generation algorithm takes the security parameter λ, and produces the public parameters PP and the secret key SK.

Encrypt (PP, M_1, ..., M_t) The encryption algorithm takes the public parameters PP and t messages, M_1, ..., M_t, and produces the corresponding ciphertext, CT_1, ..., CT_t.

Evaluate (PP, F, CT_1, ..., CT_t) Given a polynomial run time function F, and t ciphertexts, CT_1, ..., CT_t, the algorithm will perform function F onto the t ciphertexts, CT_1, ..., CT_t and produces the computed result in the ciphertext space, such that $CT^* = F((CT_1, ..., CT_t), PP)$.

Decrypt (SK, CT^*) The decryption algorithm takes the secret key, SK, and the computed result in ciphertext space, CT^*, and produces a computed result in plaintext space such that $F((CT_1, ..., CT_t), PP) = F(M_1, ..., M_t)$.

2.4 Security Models

Ring Learning with Error (R-LWE or Ring-LWE) Problem was first introduced by Lyubashevsky *et al.* [35]. The homomorphic encryption scheme in this paper uses the decision version of the R-LWE Problem as defined in the following.

Definition 6 (Decision R-LWE$_{d,q,\chi}$ Problem [19, 35]): Given the security parameter λ, let d and q be integers that depends on λ, let $R = \mathbb{Z}[x]/\langle f(x)\rangle$, where $f(x) = (x^d+1)$ and $R_q = R/_qR$. Given a distribution χ over R_q that depends on λ, the Decision R-LWE$_{d,q,\chi}$ Problem instance consists of access to an unspecified challenge oracle, $\mathcal{O}$, either a noisy pseudo-random sampler, $\mathcal{O}_s$, for random secret key, $SK \in R_q$, or truly random sampler, $\mathcal{O}_\$$. The Decision R-LWE$_{d,q,\chi}$ Problem is to distinguish the sampling between $\mathcal{O}_s$ and $\mathcal{O}_\$$, which perform respectively as follows.

$\mathcal{O}_s$: outputs noisy pseudo-random samples of the form $(a, PK) = (a, a \cdot SK + e) \in R_q \times R_q$. The element SK is drawn from uniformly random R_q, where $SK \leftarrow R_q$ and it is fixed for all samples. For each sample, the element, a is drawn from uniformly random R_q, where $a \leftarrow R_q$ and the element, e is a small error term that generated with a distribution χ, where $e \leftarrow \chi$.

$\mathcal{O}_\$$: outputs truly random samples $(a, PK) \in R_q \times R_q$ drawn independently and uniformly random in the entire domain $R_q \times R_q$.

The Decision R-LWE$_{d,q,\chi}$ Problem allows repeated queries to be sent to the challenge oracle $\mathcal{O}$. The algorithm adversary $\mathcal{A}$ decides the Decision R-LWE$_{d,q,\chi}$ Problem if $|Pr[\mathcal{A}^{O^s} = 1] - Pr[A^{O^s} = 1]|$ is non-negligible for a random $SK \in R_q$.

A selective-set model is defined for proving the security of the ABE scheme under chosen plaintext attack [34, 36]. In selective game model, the adversary is allowed to choose to be challenged on an encryption to an access structure, $\mathcal{A}^*$ and can ask for any private key, SK, such that SK does not satisfy $\mathcal{A}^*$. The formal definition of the security game is defined as follows.

Init: The adversary, $\mathcal{A}$ declares an access structure $\mathcal{A}^*$ that he wishes to be challenged upon.

Setup: The challenger runs the setup algorithm of the scheme and gives the public parameters, PP to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ is allowed to issue repeated private key queries for any attribute list L, where $L \not\models \mathcal{A}^*$. The challenger runs *keyGen* algorithm and sends a private key SK_L to $\mathcal{A}$.

Challenge: $\mathcal{A}$ submits two equal length messages, M_0 and M_1. The challenger encrypts the message with the challenge access structure $\mathcal{A}^*$. The challenger flips a random binary coin, r. If r=1, the ciphertext is given to $\mathcal{A}$, otherwise, a random element of the ciphertext space is returned.

Phase 2: Repeated ***Phase* 1**.

Guess: $\mathcal{A}$ produces a guess r' of r.

The advantage, $\in$ of adversary $\mathcal{A}$ in this game is further defined as $adv(\mathcal{A}) = |Pr[r'=r] - 1/2|$.

3. CONSTRUCTION OF THE PROPOSED CIPHERTEXT POLICY-ATTRIBUTE BASED HOMOMORPHIC ENCRYPTION (CP-ABHE$_{\text{R-LWE}}$) SCHEME

3.1 Construction of the Circuit-Based CP-ABHE$_{\text{R-LWE}}$ Scheme

The homomorphic encryption scheme used in this paper is based on the R-LWE Problem, which was introduced by Lyubashevsky *et al.* [19, 35]. The existing lattice

cryptosystems which is based on the standard Learning with Error (LWE) Problem [12, 37], suffers from the inherent quadratic overhead problem, in which each sample $(a, PK) \in R_q \times R_q$ from the R-LWE distribution is used to replace n samples $(a, PK) \in Z_q^n \times Z_q$ from the standard LWE distribution, thus reducing both public key and private key size by a factor of n [35]. This subsequently lightening the homomorphic computation, in which polynomial multiplication can be achievable with $O(n \log n)$ scalar operations and has a parallel depth of $O(\log n)$ if the multiplication is calculated by using the Fast Fourier Transform (FFT) [35].

The key question arises here is how to incorporate ABE scheme into homomorphic encryption scheme without affecting the capability of the homomorphic computation? To solve this problem, existing ABFHE scheme [26-28] only consider their access structure as a single attribute, thereby these schemes are not able to provide a "truly" fine-grained access control on these encrypted data. In practice world, data are always being encrypted with a set of attributes that describes the data access policies. To bridge this gap, we proposed a CP-ABHE$_{R\text{-}LWE}$ scheme that is able to handle a monotonic access structure over a set of attributes that well-describes the data access policies, without interfering with the homomorphic computation. First, we express a set of authorized attributes with a monotonic access structure and transform them into LSSS matrix (G, p) as discussed in Section 2.1. Second, compared to the existing schemes [26-28] that directly encoded an access structure [27] or an attribute [26, 28] within a message, we encode a message into two ciphertext components (C^0, C^1). The ciphertext C^0 is used to encode a message together with the master public key, PK_0 and a shared secret key, s, whereas the ciphertext C^1 is associated with a monotonic access structure over a set of authorized attributes. Each authorized attribute in the monotonic access structure holds a private piece of the shared secret key, s. The homomorphic computation can be conducted as usual on ciphertext C_0. For decrypting the result, only the eligible user with the corresponding set of attributes that satisfy the corresponding monotonic access structure can recover the shared secret key, s from ciphertext, C_1.

Setup $(1^\lambda, \boldsymbol{U})$ Given a security parameter, λ and a universe of attributes, $U = \{u_1, u_2, ..., u_n\}$. Choose a sufficiently large prime modulus $q = 1 \bmod (2\lambda)$, and a smaller positive integer p, where $p \ll q$ and $gcd(p, q) = 1$. Let $f(x) = (x^d+1)$, where d is a power of 2. Let $R_q = \mathbb{Z}_q[x] / \langle f(x) \rangle$ be the ring of integer polynomials modulo both $f(x)$ and q. Let $\mathcal{X} = \mathcal{X}(\lambda)$ be an error distribution over R_q and $t = \{1, 2, ..., t\}$. Select a uniformly random master secret key, $SK_0 \leftarrow R_q$, a tuple of random element, $a_t \leftarrow R_q$, and a tuple of small error terms, $e_t \leftarrow \mathcal{X}$. Define T public keys, PK_t, such that $PK_t = a_t \cdot SK_0 + pe_t \in R_q$. Next, select a root secret key SK_1 and its inverse, SK_1^{-1} from the universe of attributes U, where $SK_1, SK_1^{-1} \leftarrow \mathcal{X}$. Then, for each attribute $\{u_1, u_2, ..., u_n\}$ in U, select a pair of uniformly random $(\alpha_i, \alpha_i^{-1}) \leftarrow \mathcal{X}$, where α_i^{-1} is the inverse of α_i in R_q and a small error term, $e_i \leftarrow \mathcal{X}$. Compute $PK_i = \alpha_i \cdot SK_1 + pe_i \in R_q$. Lastly, produce the public parameters PP, and a master secret key MSK as follows.

$$PP = \{\{a_t\}_{t=1}^{t}, \{PK_t\}_{t=1}^{t}, \{PK_i\}_{i=1}^{n}\}$$
$$MSK = \{SK_0, SK_1, SK_1^{-1}, \{\alpha_i\}_{i=1}^{n}, \{\alpha_i^{-1}\}_{i=1}^{n}\}$$

Encrypt $(\boldsymbol{PP}, \mathcal{A}, \boldsymbol{M_1, ..., M_t})$ Given to the encryption function are the public parameters

PP, an access structure $\mathcal{A}$, over the universe of attributes U and t messages, $M_t \in \{1,0\}$, where $t=\{1, 2, ..., t\}$. Construct LSSS access structure (G, p) where matrix $G \in R_q^{n \times m}$, with row labels $p(i) \in U$, $\forall i \in [n]$. Then, generates a secret to be shared in the form of vector, $v = (s, r_2, ..., r_m)$, where $s \leftarrow R_q$ and $r_2, ..., r_m \leftarrow R_q$ randomly chosen. Let Gv be the vector of n shares of the secret s − according to secret sharing scheme $\prod$ over a set of attribute $U = \{u_1, u_2, ..., u_n\}$. For each i=1 to n, calculates the secret share, $\delta_i = G_i \cdot v \in R_q$, where G_i is the vector corresponding to ith row of G. Next, select a uniformly random, $r_t \leftarrow R_q$ and error terms, e'_t, $e'_{t\text{-}i} \leftarrow \mathcal{X}$. Outputs the t ciphertexts, $CT_t = (C^0_t, C^i_t)$ associated with a description of (G, p), where:

$$C^0_t = PK_t \cdot r_t \cdot s + M_t + pe'_t \in R_q$$
$$C^i_t = a_t \cdot PK_i \cdot r_t \cdot \delta_i + pe'_{t\text{-}i} \in R_q$$

Evaluate (*PP*, *F*, CT_1, ..., CT_t) Given the polynomial time computation function F, t ciphertexts, $CT_t = (C^0_t, C^j_t)$; the algorithm perform function F onto the t ciphertexts C^0_t, and produces the computed result in ciphertext space, such that $CT^* = F((C^{0*}, C^i_t), PP) = F(((C^0_1, ..., C^0_t), C^j_t), PP)$.

KeyGen (*MSK*, *A*) The key generation algorithm takes the master secret key *MSK*, and a set of attributes A, that describes the user's credentials and authorization. Then, the algorithm generates a pair of freshly random element β, and its inverse β^{-1} such that $\beta, \beta^{-1} \leftarrow \mathcal{X}$. Next, select a uniformly random error terms, e'', $e''_i \leftarrow \mathcal{X}$. The function produces a user's private key, $K = (K_0, K_i)$ for A as follows.

$$K_0 = SK_0 \cdot \beta^{-1} + pe'' \in R_q$$
$$K_i = \alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe''_i \in R_q, \forall i \in A$$

Decrypt (*K*, CT^*) Given a user's secret key K, and a computed result in ciphertext space CT^*, together with the associated access structure (G, p), the algorithm will recover the computed result in plaintext space, M^*, such that $CT^* = F((C^{0*}, C^i_t), PP) = F(((C^0_1, ..., C^0_t), C^j_t), PP) = F(M_1, ..., M_t)$. Let $A \in \mathcal{A}$ be an authorized set and $I \subset \{1, 2, ..., n\}$ defined as $I = \{i: p(i) \in A\}$. 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 s, according to G, then $\Sigma_{i \in I} \delta_i w_i = s$. Next, compute $M^{*\prime} = C^{0*}_t - K_0 \Sigma_{i \in I} C^j_t \cdot w_i \cdot K_i$ and output $M^*_t = M^{*\prime}_t \bmod p$, otherwise, outputs a false symbol $\perp$.

3.2 Construction of Non-Circuit CP-ABHE$_{\text{R-LWE}}$ Scheme

The proposed circuit-based CP-ABHE$_{\text{R-LWE}}$ scheme in Section 3.1 can be further extended to achieve Fully Homomorphic Encryption (FHE) with bootstrapping or modulus switching techniques as discussed in [12, 14, 40]. However, instead of following the existing FHE blueprint that aimed to support arbitrary computations by evaluating a deep circuit gates over binary bits, which is ideal in theory but impractical for real-world deployments, this paper further extended the circuit based CP-ABHE$_{\text{R-LWE}}$ scheme into non-circuit based approach in order to reduce the generated ciphertext size and computation complexity.

First, the technique that have been discussed in [21, 41, 42, 44] is applied here to extend the message space and represented them as a polynomial over the rings, $\mathbb{Z}_q[x]/<x^d+1>$, where d is a power of 2. In the non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme, all ciphertexts and plaintexts are encoded as a polynomial with integer coefficient less than d. Let $z \in \mathbb{Z}$ *be* an integer that represented in binary bits as circuit based approach. Then, encoding the plaintext z as the polynomial $y(x) = (\pm 1)\Sigma_{i=0}^{n} z_i x^i$, where $z_i \in \{-1, 0, 1\}$ and $n = \lfloor \log 2(|z|) \rfloor + 1$ [42]. The polynomial $y(x)$ can be decoded into plaintext z by evaluating the polynomial $y(x)$ with x=2, that is $z = y(2)$. For instance, integer 5 can be randomly presented as $x^4-x^3-x^2+1$, x^3-x-1 or x^2+1. Second, compared to the conventional bit-by-bit encoding method in the circuit-based approach that encode every single bit of plaintext as a separate ciphertext, we encode the whole plaintext as a single ciphertext, thus greatly reduced the ciphertext size and computation time. Third, the computation functions in the conventional circuit-based approach [14, 17, 18, 45] requires application to additionally convert computation functions into a deep Boolean circuits that consist of bitwise XOR and AND gates and homomorphic evaluation is executed by evaluating a deep Boolean circuits over bits. For instance, a multiplication of two 8-bit integers involving a deep Boolean circuit evaluation over a hundred bit operations with multiplication depth 16 [46], and the addition of two 8-bit integers involving an expensive carry operations on homomorphic multiplication over $\mathbb{Z}_2$ [41]. In the non-circuit based approach, both integer addition and multiplication operations in the proposed scheme are corresponding directly to the addition and multiplication of polynomials over the rings respectively. Thus leading to the reduction of the bandwidth and communication time required to transfer to a third-party service provider and more amenable to support the real-world cloud data processing. The difference between the circuit-based and non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme is further illustrated in Figs. 1 and 2 respectively.

The proposed non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme has five algorithms: ***Setup***, ***Encrypt***, ***Evaluate***, ***KeyGen*** and ***Decrypt***. The setup algorithm **(*Setup*)** and key generation algorithm (***KeyGen***) are similar to the respective algorithms found in the circuit-based CP-ABHE$_{R\text{-}LWE}$ scheme as defined in Section 3.1. The encryption algorithm **(*Encrypt*)**, the homomorphic evaluation algorithm **(*Evaluate*)** and the decryption algorithm **(*Decrypt*)** are formally defined as follows.

Encrypt (*PP*, $\mathcal{A}$, M_1, ..., M_t) The algorithm receives the public parameters, *PP*, an access structure $\mathcal{A}$, over the universe of attributes U and t messages M_t, where $t = \{1, 2, ..., t\}$. The algorithm first encode M_t as a polynomial rings $\mathbb{Z}_q[x]/<f(x)>$, denoted as *Encode*(M_t). Then the algorithm constructs the LSSS access structure (G, p) where matrix $G \in R_q^{n \times m}$, with row labels $p(i) \in U$, $\forall i \in [n]$. The algorithm generates a vector, $v = (s, r_2, ..., r_m)$, where $s \in R_q$ is the secret to be shared and $r_2, ..., r_m \leftarrow R_q$ is randomly chosen. Let Gv be the vector of n shares of the secret s, created according to the secret sharing scheme $\prod$ over a set of attribute $U = \{u_1, u_2, ..., u_n\}$. For each i=1 to n, calculates the secret share, $\delta_i = G_i \cdot v \in R_q$, where G_i is the vector corresponding to ith row of G. Next, select a uniformly random $r_t \leftarrow R_q$, and error terms, e_t', $e_{t\text{-}i}' \leftarrow \mathcal{X}$. The algorithm finally produces the t ciphertexts $CT_t = (C^0_t, C^i_t)$, associated with a description of (G, p), where:

$$C^0_t = PK_t \cdot r_t \cdot s + Encode(M_t) + pe_t' \in R_q$$
$$C^i_t = a_t \cdot PK_i \cdot r_t \cdot \delta_i + pe_{t\text{-}i}' \in R_q$$

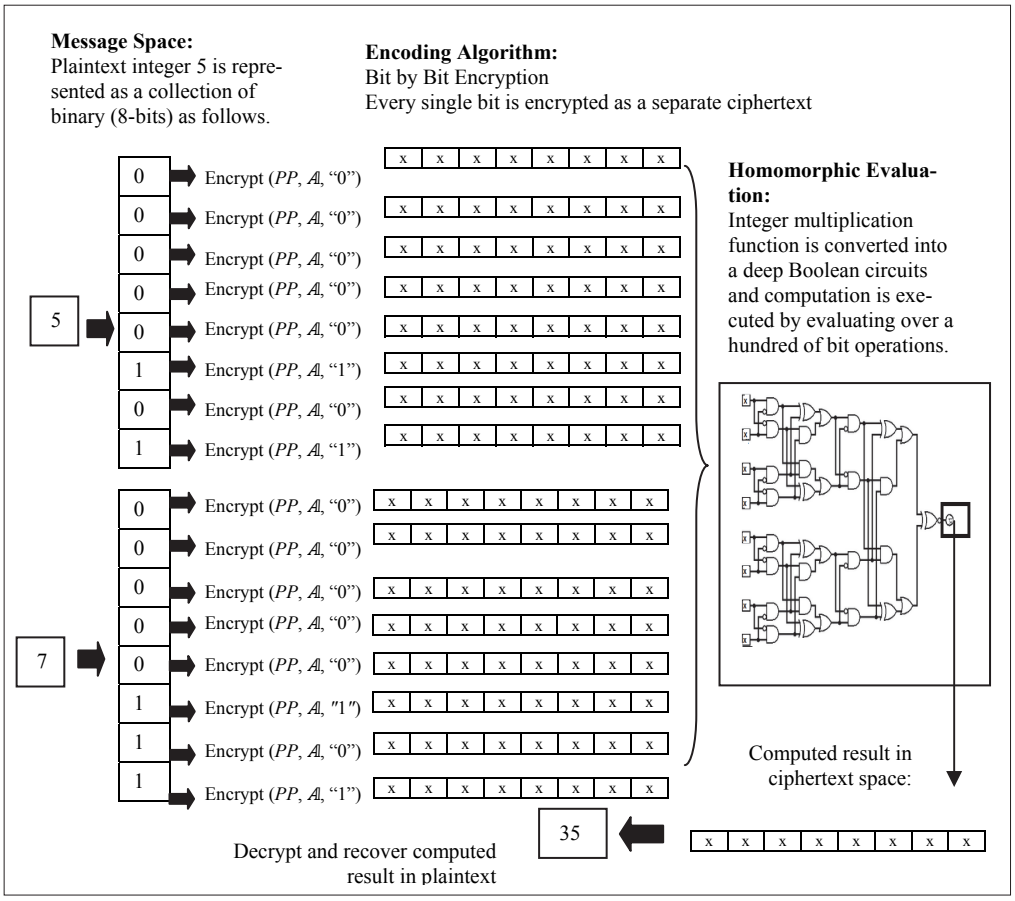

Fig. 1. CP-ABHE$_{R\text{-}LWE}$ scheme constructed based on the conventional circuit-based approach.

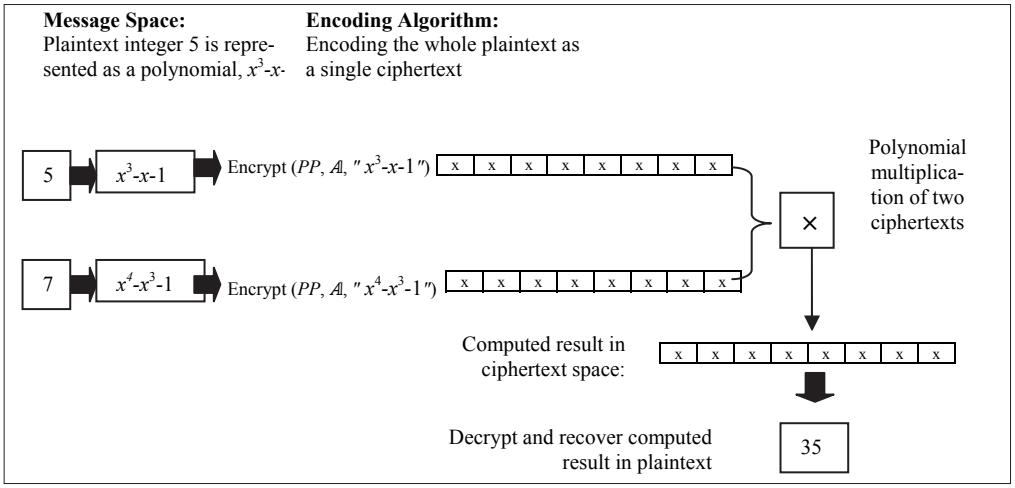

Fig. 2. Extending the circuit-based CP-ABHE$_{R\text{-}LWE}$ scheme into a non-circuit-based approach.

Evaluate (PP, F, CT_1, ..., CT_t) Given a polynomial time computation function F, that consists a tuple of two fundamental operations: addition and multiplication, t ciphertexts, $CT_t = (C_t^0, C_t^i)$, the algorithm applies function F onto the t ciphertexts, C_t^0. Homomorphic addition is done by a simple coefficient-wise addition of t ciphertext C_t^0. Homomorphic multiplication is corresponding to the multiplication of two ring polynomials $\mathbb{Z}_q[x]/<x^d+1>$. The algorithm produces the computed result in ciphertext space, CT^*, such that $CT^* = F((C_t^{0*}, C_t^i), PP) = F(((C_1^0, ..., C_t^0), C_t^i), PP)$.

Decrypt (K, CT^*) Given the user's secret key K, and the computed result in ciphertext space CT^*, together with associated access structure (G, p), the algorithm recovers the computed result in plaintext space, M_t^*, such that $CT^* = F((C_t^{0*}, C_t^i), PP) = F(((C_1^0, ..., C_t^0), C_t^i), PP) = F(M_1, ..., M_t)$. Let $A \in \mathbb{A}$ be an authorized set and $I \subset \{1, 2, ..., n\}$ defined as $I = \{i: p(i) \in A\}$. 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, s according to G, then $\Sigma_{i\in I}\, \delta_i w_i = s$. Next, compute $M_t^{*}{''} = C_t^{0*} - K_0 \Sigma_{i\in I}\, C_t^i . w_i . K_i$ and output $M_t^{*}{'} = M_t^{*}{''} \bmod p$, otherwise, outputs a false symbol $\perp$. Lastly, the algorithm decodes the computed result in plaintext space, such that $M_t^* = Decode\,(M_t^{*}{'})$.

Correctness The correctness of the proposed CP-ABHE$_{R\text{-}LWE}$ scheme follows the correctness of LSSS linear reconstruction property [30]. Let $A \in \mathbb{A}$ be an authorized set, and $I \subset \{1, 2, ..., n\}$ defined as $I = \{i: p(i) \in A\}$, then there exists constants $\{w_i \in R_q\}_{i\in I}$ and $\Sigma_{i\in I}\, \delta_i w_i = s$. The proof of the correctness is further described as follows.

$$
\begin{aligned}
M_t'' &= C^{0*}{}_t - K_0 \cdot \Sigma_{i\in I}\, C^i{}_t \cdot w_i \cdot K_i \\
&= C^{0*}{}_t - K_0 \cdot \Sigma_{i\in I}\, (a_t \cdot PK_i \cdot r_t \cdot \delta_i + pe_{t\text{-}i}') \cdot w_i \cdot K_i \\
&= C^{0*}{}_t - K_0 \cdot \Sigma_{i\in I}\, (a_t \cdot r_t \cdot s \cdot PK_i \cdot K_i) - p\Sigma_{i\in I}\, e_{t\text{-}i}' \cdot w_i \cdot K_i \\
&= C^{0*}{}_t - K_0 \cdot \Sigma_{i\in I}\, a \cdot r_t \cdot s\, (\alpha_i \cdot SK_1 + pe_i)\, (\alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe_i'') - p\Sigma_{i\in I}\, e_{t\text{-}i}' \cdot w_i \cdot K_i \\
&= (PK_t \cdot r_t \cdot s + Encode(M_t) + pe_t') - K_0 \cdot a_t \cdot r_t \cdot s \cdot \beta - pK_0 \Sigma_{i\in I}\, e_i'' \cdot \alpha_i \cdot SK_1 - pK_0 \Sigma_{i\in I} \\
&\quad e_i(\alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe_i'') - p\Sigma_{i\in I}\, e_{t\text{-}i}' \cdot w_i \cdot K_i \\
&= ((a_t \cdot SK_0 + pe_t)\, r_t s + Encode(M_t) + pe_t') - (SK_0 \cdot \beta^{-1} + pe'')\, (a_t \cdot r_t \cdot s \cdot \beta) - \\
&\quad pK_0 \Sigma_{i\in I}\, e_i'' \alpha_i \cdot SK_1 - pK_0 \Sigma_{i\in I}\, e_i (\alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe_i'') - p\Sigma_{i\in I}\, e_{t\text{-}i}' \cdot w_i \cdot K_i \\
&= (a_t SK_0 r_t s + pe_t r_t s + Encode(M_t) + pe_t') - a_t SK_0 r_t s + p\,(e'' a_t r_t s\beta) - \\
&\quad pK_0 \Sigma_{i\in I}\, e_i'' \alpha_i SK_1 - pK_0 \Sigma_{i\in I}\, e_i (\alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe_i'') - p\Sigma_{i\in I}\, e_{t\text{-}i}' w_i K_i \\
&= Encode(M_t) + pe_t r_t s + pe_t' + p\,(e'' \cdot a_t \cdot r_t \cdot s \cdot \beta) - pK_0 \Sigma_{i\in I}\, e_i'' \alpha_i SK_1 - \\
&\quad pK_0 \Sigma_{i\in I}\, e_i (\alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe_i'') - p\,\Sigma_{i\in I}\, e_{t\text{-}i}' w_i K_i
\end{aligned}
$$

$$
\begin{aligned}
M_t' &= M_t'' \bmod p = Encode(M_t) \\
M_t &= Decode\,(M_t')
\end{aligned}
$$

Similar to most of the lattice based encryption schemes, the encryption algorithm of the proposed CP-ABHE$_{R\text{-}LWE}$ scheme involves adding a noise terms into the ciphertexts. Therefore, for ensuring the correctness of the decryption algorithm, overall noise terms (e, e_i, e_t', e_i', e'', e_i'') added to the ciphertext must be small enough compared to the ratio q to p, denote as $\Delta = \lfloor q/p \rfloor$ and the choice of e, e_i, e_t', e_i', e'', e_i'', s, t, q, p, r_t, w_i, K_i, SK_1 and SK_1^{-1} should satisfies $\|M''\|_{\infty^*} < \Delta/2$.

Computation on Encrypted Cloud Data The computation on the encrypted cloud data is implemented by using the evaluation algorithm, ***Evaluate* (*PP*, *F*, CT_1, ..., CT_t).** Let ***F*** be the cloud data computation function (*e.g.* means, standard deviation, logistic regression, *etc.*) that consists a tuple of homomorphic addition f_{add},and multiplication operation f_{mult}. Let CT_1 and CT_2 are fresh ciphertexts that had been generated by the Encryption algoritiom, ***Encrypt***, under the same access structure (G, p), such that $CT_1 = (C^0_1, C^i_1)$ and $CT_2 = (C^0_2, C^i_2)$. As mentioned in Section 3.1, the homomorphic evaluation of the proposed CP-ABHE$_{R\text{-}LWE}$ scheme only involves the component of ciphertext, C^0_i. The homomorphic addition f_{add}, is calculated by using a simple coefficient-wise addition of two ciphertext over the R_q as shown in the following.

$$\begin{aligned} f_{add}(CT_1, CT_2) &= C^0_1 + C^0_2 \\ &= (PK_1\, r_1\, s + Encode(M_1) + pe_1') + (PK_2\, r_2\, s + Encode(M_2) + pe_2') \\ &= Encode(M_1) + Encode(M_2) + s\,(PK_1\, r_1 + PK_2\, r_2) + p\,(e_1' + e_2') \end{aligned}$$

Meanwhile, the homomorphic multiplication f_{mult}, is implemented by multiplying two polynomials over R_q as shown below.

$$\begin{aligned} f_{mult}(CT_1, CT_2) &= C^0_1 \times C^0_2 \\ &= (PK_1 r_1 s + Encode(M_1) + pe_1') \times (PK_2\, r_2\, s + Encode(M_2) + pe_2') \\ &= Encode(M_1) \times Encode(M_2) + s^2(PK_1 r_1 PK_2.r_2) + \\ &\quad s(PK_1\, r_1 Encode(M_2) + PK_2 r_2 Encode(M_1)) + pe_1'(PK_2 r_2 + \\ &\quad Encode(M_2)) + pe_2'(PK_1 r_1 + Encode(M_2)) + pe_1'\, pe_2' \end{aligned}$$

The computed results, from the homomorphic addition $f_{add}(CT_1, CT_2)$, and homomorphic multiplication $f_{mult}(CT_1, CT_2)$, can be recovered by using the decryption algorithm, (***Decrypt***); excepts the decryption key in the homomorphic multiplication $f_{mult}(CT_1, CT_2)$, which is gather by joining the secret keys of CT_1 and CT_2. The decryption key is defined as $SK^2 s^2$. For certain outsourced cloud data computation, it might be unacceptable for the decryption key (for $f_{mult}(CT_1, CT_2)$) to depend on the number of multiplication operations being evaluated. To avoid such cases, the re-linearization technique [12, 41] can be applied here to transform and switch the secret key of the computed ciphertext after every multiplication operation.

Besides that, the noise terms of $f_{mult}(CT_1, CT_2)$ grows quadractically, which is larger than $f_{add}(CT_1, CT_2)$. To ensure the computed result can be recoverred correctly, after performing t multiplications, the message M must be encoded as a polynomial of degree at most d/t [41, 42]. For handling the outsourced cloud data computations such as means, variance, standard deviation, predictive analysis and other statistical analysis, it is an acceptable trade-off since such computation generally involves only single multiplication or at most, a limited number of multiplications.

Intuitively, it assumes that the homomorphic computation can be conducted if both ciphertexts had been encrypted under the same monotonic access structure as defined in Section 2.1. Any superset of the set A satisfying the access structure $\mathcal{A}$ is also able to recover the computed result. Let A be a set of attributes satisfying an access structure $\mathcal{A}$, and if exit B such that $A \subseteq B$ the B will also satisfies $\mathcal{A}$. For instance, consider a simple monotonic access structure $\mathcal{A} = \{u_1 \cap u_2\}$, if $A=\{u_1, u_2\}$ satisfying an access structure $\mathcal{A}$, then $B=\{u_1, u_2, u_3\}$ will also satisfying $\mathcal{A}$.

4. SECURITY ANALYSIS

The security of the proposed CP-ABHE$_{R\text{-}LWE}$ scheme is constructed based on the hardness of R-LWE Problem. This section shows that the CP-ABHE$_{R\text{-}LWE}$ scheme is secure under a selective-set model with the hardness of Decision R-LWE$_{d,q,\chi}$ Problem as in Definition 6, Section 2.3.

Theorem 1: If there exists a Probabilistic Polynomial Time (PPT) algorithm adversary, $\mathcal{A}$ with the advantage $\in$ in selective-set model for the CP-ABE$_{R\text{-}LWE}$ scheme as construction above, then there exists a PPT algorithm simulator, $\mathcal{B}$ that decides the Decision R-LWE$_{d,q,\chi}$ Problem with advantage $\in/2$.

***Proof*:** As described in Definition 6, the Decision R-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 $SK_0 \leftarrow R_q$, or truly random $\mathcal{O}_\$$. Then, the simulator $\mathcal{B}$ uses adversary $\mathcal{A}$ to differentiate oracle $\mathcal{O}$. Firstly, $\mathcal{B}$ queries the R-LWE oracle for (t+1) times and receives two fresh pairs $(w_k, v_k) \leftarrow R_q$ and $(x_k, y_k) \leftarrow \mathcal{X}$ where $k \in \{0,1, 2, ..., t\}$. Then, $\mathcal{B}$ continues as the following.

Init. Given a universe of attributes $U = \{u_1, u_2, ..., u_n\}$, $\mathcal{A}$ declares access structure $\mathbb{A}^*$ that he wishes to be challenged upon and announces his intention to $\mathcal{B}$.

Setup: $\mathcal{B}$ runs the Setup Algorithm of the proposed CP-ABE$_{R\text{-}LWE}$ scheme and construct the public parameter, PP as follows.

- Define $PK_t = pw_t \in R_q$.
- For each $i \in U$, define $PK_i = px_i \in R_q$ if $i \in \mathbb{A}^*$; otherwise, define $PK_i = \alpha_i \cdot SK_1 + pe_i \in R_q$ as shown in Section 3.1.

Next, $\mathcal{B}$ returns the PP = $\{\{a_t\}_{t=1}^{t}, \{PK_t\}_{t=1}^{t}, \{PK_i\}_{i=1}^{n}\}$ to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ sends private key queries for an attribute list $A^* = \{A^*_1, A^*_2, ..., A^*_j\}$, where $A^* \not\models \mathbb{A}^*$ for all i. $\mathcal{B}$ runs ***KeyGen*** algorithm of the proposed scheme to construct private key K, for A^* as follows.

$$K_0 = SK_0 \cdot \beta^{-1} + pe'' \in R_q$$
$$K_i = \alpha_i^{-1} \cdot SK_1^{-1} \cdot \beta + pe_i'' \in R_q, \forall i \in A^*$$

Challenge: $\mathcal{A}$ signals that he is ready to accept challenges and sends a challenge message, M to $\mathcal{B}$. $\mathcal{B}$ flips a fairly binary coin r, and generate challenge ciphertext that encrypted under the access structure $\mathbb{A}^*$ as follows.

- If $r = 0$, $\mathcal{B}$ randomly choose $z_0 \leftarrow R_q$, $z_i \leftarrow \mathcal{X}$ and set $C_0 = pz_0 \in R_q$ and $C_i = pz_i \in R_q$.
- If $r = 1$, $\mathcal{B}$ define $C^0_t = pv_t + M \in R_q$ and $C_i = py_i \in R_q$ for $i \in \mathbb{A}^*$.

Phase 2: Repeated ***Phase*** 1.

Guess: $\mathcal{B}$ receives a guess r' for r from $\mathcal{A}$, and outputs r' as the answer to the R-LWE challenge. If ($r' = r$), output "$\mathcal{O}' = \mathcal{O}_s$", otherwise, output "$\mathcal{O}' = \mathcal{O}_\$$".

From the definition of selective-set model [47, 48], the advantage of $\mathcal{A}$ is defined as adv($\mathcal{A}$) = $|Pr[r'=r] - 1/2|$. Therefore, when the R-LWE$_{d,q,\chi}$ the oracle $\mathcal{O}$ is:

- a noisy pseudo-random $\mathcal{O}_s$: $\mathcal{A}$ has an advantage $\in$, then $|Pr[r'=r \mid \mathcal{O}=\mathcal{O}_s]| = 1/2 + \in$ and $|Pr[\mathcal{O}'=\mathcal{O} \mid \mathcal{O}=\mathcal{O}_s]| = 1/2 + \in$.
- a truly random $\mathcal{O}_\$$: $\mathcal{A}$ has no advantage ϵ and has no information regarding the r, then $|Pr[r'\neq r \mid \mathcal{O}=\mathcal{O}_\$]| = 1/2$ and $|Pr[\mathcal{O}'' = \mathcal{O}' \mid \mathcal{O}=\mathcal{O}_\$]| = 1/2$.

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

$$1/2|Pr[\mathcal{O}'' = \mathcal{O} \mid \mathcal{O}= \mathcal{O}_s]|+1/2|Pr[\mathcal{O}''= \mathcal{O} \mid \mathcal{O}= \mathcal{O}_\$]| - 1/2$$
$$= 1/2\,(1/2 + \in) + 1/2(1/2) - 1/2$$
$$= \in/2$$

Thus, this proof that there exists a PPT algorithm simulator $\mathcal{B}$, that decides the Decision R-LWE$_{d,q,\chi}$ Problem with advantage $\in/2$.

5. PERFORMANCE ANALYSIS

In this section, the performance of the proposed CP-ABHE$_{\text{R-LWE}}$ scheme is evaluated against the existing ABHE schemes: Gentry's Scheme [26], PBHE scheme [27] and IBFHE scheme [28]. The comparison result is summarized in Table 1. Similar with the existing ABHE schemes [26-28], the proposed CP-ABHE$_{\text{R-LWE}}$ scheme is constructed based on the hardness of the lattice problem. The proposed CP-ABHE$_{\text{R-LWE}}$ scheme and PBHE scheme [27] are constructed based on a more efficient lattice problem – R-LWE Problem, thus it enjoys a faster computation and smaller ciphertext and public key size as compared to the Gentry's Scheme [26] and the IBFHE scheme [28], which are still suffering from the inherent quadratic overhead of the LWE Problem.

Both Gentry's scheme [26] and IBFHE scheme [28] construct the ABHE scheme by incorporating the Key Policy-Attribute Based Encryption (KP-ABE) scheme into homomorphic encryption; whereas, the proposed CP-ABHE$_{\text{R-LWE}}$ scheme and PBHE scheme [27] embedded the CP-ABE (Ciphertext Policy-Attribute Based Encryption) into homomorphic encryption by encrypting a message with an access control policies. The main advantage of using CP-ABE scheme here is that, the data can be encrypted prior to having the knowledge of the authorized users set. Therefore, the proposed CP-ABHE$_{\text{R-LWE}}$ scheme and PBHE scheme [26] is better in supporting scalability compared to the Gentry's scheme [26] and the IBFHE scheme [28].

The key issue in designing ABHE scheme is to maintain the homomorphic evaluation capability. To achieve this, both the Gentry's scheme [26] and the PBHE scheme [27] only consider a single attribute access structure and therefore their homomorphic evaluation is limited to the ciphertexts that had been encrypted under the same attribute. Meanwhile, the IBFHE scheme [28] is capable to support homomorphic evaluation on ciphertext that had been encrypted with different attributes by inheriting a multi-key switching concept from [15]. However, their access structure is limited to a single attribute (or a set of "sub-attributes"), therefore limiting their applicability in supporting real-

Table 1. Functionality comparison of the proposed CP-ABHE$_{R\text{-}LWE}$ with existing works.

	Gentry's Scheme [26]	PBHE Scheme [27]	IBFHE Scheme [28]	Proposed CP-ABHE$_{R\text{-}LWE}$ Scheme
Security Assumptions	LWE Problem	R-LWE Problem	LWE Problem	R-LWE Problem
Embedded ABE Scheme	KP-ABE	CP-ABE	KP-ABE	CP-ABE
Access Structure	Single Attribute	Single Attribute	Single Attribute	Monotonic Access Structure over a set of attributes
Homomorphic Computation Capacity	Single Attribute	Single Attribute	Different Attributes	Different Attributes
Encoding Algorithm	Circuit Based Approach	Circuit Based Approach	Circuit Based Approach	Non-Circuit Based Approach

istic scenes. Meanwhile, the proposed CP-ABHE$_{R\text{-}LWE}$ scheme is capable to handle a monotonic access structure over a set of attributes without affecting the homomorphic evaluation capability as discussed in Section 3.1.

While existing ABHE schemes [26-28] are constructed based on circuit based approach that aimed to support arbitrary computation functions, the circuit based CP-ABHE$_{R\text{-}LWE}$ scheme can be further extended into non-circuit based approach in order to support real-world cloud data processing. In order to assess the practical efficiency of these schemes, we implemented and compared the R-LWE based ABHE schemes: PBHE scheme [27], the proposed circuit-based CP-ABHE$_{R\text{-}LWE}$ and non-circuit based CP-ABHE$_{R\text{-}LWE}$ in terms of their computation, storage and communication efficiency. The experiment is implemented with C++ by using FLINT 2.5.2 and GMP 6.1. The implementation of PBHE scheme [27] are further optimized with the scale-invariant technique as discussed in [54, 55]. The experiments are evaluated with 80-bit security level, in which the setting of parameters, n = 4096 bits, $\lceil \log(q) \rceil$ = 192 bits, standard deviation, σ of the error distribution, $\mathcal{X}$ = 8 and the 32-bit integers are randomly selected as input data. The implementation result is summarized in Table 2. Measurements were recorded from a machine with the following specification: Intel(R) Core(TM)i7-3612M CPU @ 2.10 GHz 8GB RAM, running in 64-bit mode and the values shown are the mean of 100 measurements of the respective operations.

As shown in Table 2, the proposed circuit based CP-ABHE$_{R\text{-}LWE}$ scheme enjoys a higher efficiency in both of the *KeyGen* and *Encrypt* algorithms as compared to optimized PBHE scheme [27]. In terms of homomorphic addition and multiplication operations, storage and communication efficiency, the optimized PBHE scheme [27] is outperformed the proposed circuit based CP-ABHE$_{R\text{-}LWE}$ scheme, as the optimized PBHE only consists of a single ring components. However, with the non-circuit based approach, the computation time and ciphertext size of the proposed CP-ABHE$_{R\text{-}LWE}$ scheme are dramatically reduced. For instance, with the 32-bit of data, ciphertext size can be reduced up to 32 times, compared to the circuit-based CP-ABHE$_{R\text{-}LWE}$ scheme. Therefore, the proposed non-circuit based CP-ABHE$_{R\text{-}LWE}$ scheme is better suited to support real-world outsourced cloud data computations as compared to existing ABHE schemes [26-28].

Table 2. Practical efficiency comparison of the proposed CP-ABHE$_{R\text{-}LWE}$ with existing R-LWE based ABHE scheme.

	PBHE Scheme [27]	Proposed Circuit Based CP-ABHE$_{R\text{-}LWE}$	Proposed Non-Circuit Based CP-ABHE$_{R\text{-}LWE}$
Computation Efficiency			
i) KeyGen (s)	1.900	0.147	0.035
ii) Encrypt (s)	0.774	0.1609	0.068
iii) Decrypt (s)	0.555	0.651	0.019
iv) Add (s)	0.046	0.102	0.002
v) Mult (s)	21.731	80.027	0.073
Storage and Communication Efficiency			
i) Ciphertext Size (KB)	3765	7530	236

6. CONCLUSIONS

This paper aimed to solve the open problem of constructing ABHE scheme – that is how to incorporate ABE scheme into homomorphic encryption scheme without affecting the homomorphic evaluation capabilities. While existing ABHE schemes are limited to express their access structure as a single attribute or use a single attribute to represent a set of "sub-attributes", the proposed CP-ABHE$_{R\text{-}LWE}$ is capable to handle a monotonic access structure over a set of authorized attributes without affecting the homomorphic evaluation capabilities. To be applicable to the real-world outsourced cloud data processing environments, the proposed CP-ABHE$_{R\text{-}LWE}$ scheme is further extended into non-circuit based approach. Subsequently, this paper proved that the proposed CP-ABHE$_{R\text{-}LWE}$ scheme is secure under a selective-set model with the hardness of Decision R-LWE$_{d,q,x}$ Problem and resists against collusion attacks. For future works, how the CP-ABHE$_{R\text{-}LWE}$ scheme can be further extended to support homomorphic evaluation across a different set of monotonic access structure is another interesting research topic. For instance, in some special circumstances, a public listed organization might want to predict financial risks based on the data that being collected from their subsidiary companies that was encrypted with different data access policies; or a research agency that conduct a predictive analysis on disease outbreak might need the data inputs from different medical and healthcare centers. A current trivial approach requires organizations to download all encrypted outsourced cloud data from cloud service providers, decrypting them and re-encrypt them again with the new secret key $SK_{0'}$, and new shared secret key s', under the new monotonic access structure $\mathcal{A}'$, will be created. Such approach is clearly cumbersome and problematic. While the proposed CP-ABHE$_{R\text{-}LWE}$ scheme aimed to protect security and privacy issues of outsourced cloud data computation, enterprises are concerned about the accuracy of the computed results, and whether these un-trusted service providers are performing their actual work correctly or simply returning plausible results [50]. In future, the proposed CP-ABHE$_{R\text{-}LWE}$ scheme can be further investigated to work with Verifiable Computation (VC) in order to guarantee its verifiable results.

REFERENCES

1. C. Gentry, "A fully homomorphic encryption scheme," Ph.D. Thesis, Department of Computer Science, Standford University, 2009.
2. N. P. Smart and F. Vercauteren, "Fully homomorphic encryption with relatively small key and ciphertext sizes," in *Proceedings of International Conference on Pub-c Key Cryptography*, LNCS, Vol. 6056, 2010, pp. 420-43.
3. C. Gentry and S. Halevi, "Fully homomorphic encryption without squashing using depth-3 arithmetic circuits," in *Proceedings of the 52nd IEEE Annual Symposium on Foundations of Computer Science*, 2011, pp. 107-109.
4. C. Gentry and S. Halevi, "Implementing Gentry's fully-homomorphic encryption scheme," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS, Vol. 6632, 2011, pp. 129-148.
5. M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Proceedings of Advances in Cryptology – EURO-CRYPT*, LNCS, Vol. 6110, 2010, pp. 24-43.
6. J-S. Coron, D. Naccache, and M. Tibouchi, "Public key compression and modulus switching for fully homomorphic encryption over the integers," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS, Vol. 7237, 2012, pp. 446-464.
7. J. H. Cheon, J. Coron, J. Kim, and M. Lee, "Batch fully homomorphic encryption over the integers," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS, Vol. 7881, 2013, pp. 315-335.
8. J. Kim, M. Lee, A. Yun, and J. Cheon, "CRT-based fully homomorphic encryption over the integers," *Information Sciences*, Vol. 310, 2015, pp. 149-162.
9. L. Zhang and Q. Yue, "A fast integer-based batch full-homomorphic encryption scheme over finite field," *IACR Cryptology ePrint Archive*, No. 2013/793, pp. 1-11. https://eprint.iacr.org/2013/793.
10. J. Coron, T. Lepoint, and M. Tibouchi, "Scale-invariant fully homomorphic encryption over the integers," in *Proceedings of Public-Key Cryptography*, LNCS, Vol. 8383, 2014, pp. 311-328.
11. K. Nuida and K. Kurosawa, "(Batch) Fully homomorphic encryption over integers for non-binary message spaces," in *Proceedings Advances in Cryptology – of EU-ROCRYPT*, LNCS, Vol. 9056, 2015, pp. 537-555.
12. Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proceedings of the 52nd IEEE Annual Symposium on Foundations of Computer Science*, 2011, pp. 97-106.
13. Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(Leveled) Fully homomorphic encryption without bootstrapping," *ACM Transactions on Computation Theory* (*TOCT*) – *Special Issue on Innovations in Theoretical Computer Science*, Part II, Vol. 6, 2014, Article 13.
14. C. Gentry, S. Halevi, and N. P. Smart, "Better bootstrapping in fully homomorphic encryption," in *Proceedings of International Conference on Public Key Cryptog-raphy*, LNCS, Vol. 7293, 2012, pp. 1-16.
15. A. López-Alt, E. Tromer, and V. Vaikuntanathan, "On-the-fly multiparty computa-tion on the cloud via multikey fully homomorphic encryption," in *Proceedings of the 44th ACM Symposium on Theory of Computing*, 2012, pp. 1219-1234.

16. D. Stehlé and R. Steinfeld, "Faster fully homomorphic encryption," in *Proceedings of Advanced Cryptology – ASIACRYPT*, LNCS, Vol. 6477, 2010, pp. 377-394.
17. W. Zhang, S. Liu, and Y. Xiaoyuan, "RLWE-based homomorphic encryption and private information retrieval," in *Proceedings of the 5th Intelligent Networking and Collaborative Systems*, 2013, pp. 535-540.
18. Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-lwe and security for key dependent messages," in *Advances in Cryptology – CRYPTO*, LNCS, Vol. 6841, 2011, pp. 505-524.
19. V. Lyubashevsky, C. Peikert, and O. Regev, "A toolkit for ring-LWE cryptography," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS, Vol. 7881, 2013, pp. 35-54.
20. H. Zhou and G. Wornell, "Efficient homomorphic encryption on integer vectors and its applications," in *Proceedings of Information Theory and Applications Workshop*, 2014, pp. 1-9.
21. Fujitsu Laboratories Ltd., "Fujitsu develops world's first homomorphic encryption technology that enables statistical calculations and biometric authentication," *Fujitsu Online Press Release Archive*, http://www.fujitsu.com/global/about/resources/news/press-releases/2013/0828-01.html, 2013.
22. J. Sen, "Homomorphic encryption – theory and application," *Theory and Practice of Cryptography and Network Security Protocols and Technologies*, INTECH Publishers, 2013, pp. 1-32.
23. Y. Hu, "Improving the efficiency of homomorphic encryption schemes," Ph.D. Thesis, Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, 2013.
24. Y. Doröz, A. Shahverdi, T. Eisenbarth, and B. Sunar, "Toward practical homomorphic evaluation of block ciphers using prince, in *Proceedings of Financial Cryptography and Data Security Workshop*, LNCS, Vol. 8438, 2014, pp. 208-220.
25. Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "Fully homomorphic encryption without bootstrapping," in *Proceedings of the 3rd Innovations in Theoretical Computer Science*, 2012, pp. 309-325.
26. C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based," in *Proceedings of Advances in Cryptology – CRYPTO*, LNCS, Vol. 8042, 2013, pp. 75-92.
27. M. Clear and C. McGoldrick, "Policy-based non-interactive outsourcing of computation using multikey FHE and CP-ABE," in *Proceedings of the 10th Security and Cryptography*, 2013, pp. 444-452.
28. M. Clear and C. McGoldrick, "Bootstrappable identity-based fully homomorphic encryption," in *Proceedings of International Conference on Cryptology and Network Security*, LNCS, Vol. 8831, 2014, pp. 1-19.
29. ISO IEC 27000 2014, Information Security Definitions, 2014.
30. A. Beimel, "Secure schemes for secret sharing and key distribution," Ph.D. Thesis, Israel Institute of Technology, Haifa, Israel, 1996.
31. B. Balamurugan and P. V. Krishna, "Extensive survey on usage of attribute based encryption in cloud," *Journal of Emerging Technologies in Web Intelligence*, Vol. 6, 2014, pp. 263-272.
32. L. Pang, J. Yang, and Z. Jiang, "A survey of research progress and development

tendency of attribute-based encryption," *The Scientific World Journal*, Vol. 2014, 2014, pp. 1-14.
33. C. Lee, P. Chung, and M. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal Network Security*, Vol. 15, 2013, pp. 231-240.
34. B. Waters, "Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization," in *Proceedings of International Conference on Public Key Cryptography*, LNCS, Vol. 6571, 2011, pp. 53-70.
35. V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS, Vol. 6110, 2010, pp. 1-23.
36. Y. Wang, "Lattice ciphertext policy attribute-based encryption in the standard model," *International Journal Network Security*, Vol. 16, 2014, pp. 358-365.
37. Z. Chen, J. Wang, L. Chen, and X. Song, "A Regev-type fully homomorphic encryption scheme using modulus switching," *The Scientific World Journal*, Vol. 2014, 2014, pp. 1-12.
38. D. Koo, J. Hur, and H. Yoon, "Secure and efficient data retrieval over encrypted data using attribute-based encryption in cloud storage," *Journal Computers and Electrical Engineering*, Vol. 39, 2013, pp. 34-46.
39. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Computer and Communications Security*, 2006, pp. 89-98.
40. C. Gentry, S. Halevi, and N. Smart, "Fully homomorphic encryption with polylog overhead," *Lecture Notes in Computer Science*, LNCS, Vol. 7237, 2012, pp. 465-482.
41. M. Naehrig, K. Lauter, and V. Vaikuntanathan, "Can homomorphic encryption be practical?" in *Proceedings of the 3rd ACM Workshop on Cloud Computing Security*, 2011, pp. 113-124.
42. J. W. Bos, K. Lauter, and M. Naehrig, "Private predictive analysis on encrypted medical data," *Journal of Biomedical Informatics*, Vol. 50, 2014, pp. 234-243.
43. W. L. Chang, "NIST big data interoperability framework: Volume 4, security and privacy," *Special Publication* (*NIST SP*) 1500-4, 2015.
44. A. Costache and N. P. Smart, "Which ring based somewhat homomorphic encryption scheme is best?" in *Proceedings of Topics in Cryptology*, LNCS, Vol. 9610, 2016, pp. 325-340.
45. Z. Brakerski, "Fully homomorphic encryption without modulus switching from classical GapSVP," in *Proceedings of CRYPTO*, LNCS, Vol. 7417, 2012, pp. 868-886.
46. C. Aguilar-Melchor, S. Fau, C. Fontaine, G. Gogniat, and R. Sirdey, "Recent advances in homomorphic encryption: A possible future for signal processing in the encrypted domain," *IEEE Signal Processing Magazine*, Vol. 30, 2013, pp. 108-117.
47. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, 2006, pp. 89-98.
48. W. Zhu, J. Yu, W. Ting, Z. Peng, and W. Xie, "Efficient attribute-based encryption from R-LWE," *Chinese Journal of Electronic*, Vol. 23, 2014, pp. 778-782.
49. Z. Brakerski, C. Gentry, and S. Halevi, "Packed ciphertexts in LWE-based homomorphic encryption," in *Proceedings of International Conference on Public-Key*

Cryptography, LNCS, Vol. 7778, 2013, pp. 1-13.
50. S. F. Tan and A. Samsudin, "A survey of homomorphic encryption for outsourced big data computation," *KSII Transactions on Internet and Information Systems*, Vol. 10, 2016, pp. 3826-3851.
51. Z. Chen and X. Song, "A multi-bit fully homomorphic encryption with shorter public key from LWE," *IACR Cryptoly ePrint Archive*, Report 2015/1143, 2015.
52. Y. Ding, X. Li, H. Lu, and X. Li, "A novel fully homomorphic encryption based on LWE," *Wuhan University Journal of Natural Sciences*, Vol. 21, 2016, pp. 84-92.
53. K. Rohloff and D. B. Cousins, "A scalable implementation of fully homomorphic encryption built on NTRU," in Proceedings of International Conference on Financial Cryptography and Data Security, LNCS, Vol. 8438, 2014, pp. 221-234.
54. T. Lepoint and M. Naehrig, "A comparison of the homomorphic encryption schemes FV and YASHE," in *Proceedings of AFRICACRYPT*, LNCS, Vol. 8469, 2014, pp. 318-335.
55. J. W. Bos, K. Lauter, J. Loftus, and M. Naehrig, "Improved security for a ring-based fully homomorphic encryption scheme," *Cryptography and Coding*, LNCS, Vol. 8308, 2013, pp. 45-64.

Soo-Fun Tan (陳淑芬) is a Ph.D. candidate at the School of Computer Sciences, Universiti Sains Malaysia (USM). She received her Bachelor of Information Technology (majoring in E-Commerce) and Master of Science (Computer Science) from Universiti Malaysia Sabah (UMS) in 2006 and 2009 respectively. Previously, she worked as a Lecturer at the School of Engineering and Information Technology, UMS. Her research interests include cryptography, information and network security. She has published over 30 papers, which include book chapters, journals, technical reports and proceedings, as well as received research grants in the related fields. She is a member of Information Security Professional Malaysia (ISPA) and International Association of Computer Science and Information Technology (IACSIT). She is also an IBM Certified Academic Associate, a certified IPv6 Network Engineer and a certified IPv6 Security Engineer.

Azman Samsudin is a Professor at the School of Computer Science, Universiti Sains Malaysia (USM). He earned his B.Sc. in Computer Science from University of Rochester, New York, USA, in 1989. Later, he received his M.Sc. and Ph.D. in Computer Science, in 1993 and 1998, respectively, both from the University of Denver, Colorado, USA. Recently, he serves as Deputy Dean at the Institute of Postgraduate Studies, USM. His research interests include cryptography, switching networks and parallel computing. He has published more than 150 articles over a series of books, professional journals and conferences.