cryptography

MDPI

Article

An Efficient Tate Pairing Algorithm for a Decentralized Key-Policy Attribute Based Encryption Scheme in Cloud Environments

Balaji Chandrasekaran * and Ramadoss Balakrishnan

Department of Computer Applications, National Institute of Technology, Tiruchirappalli 620015, India; brama@nitt.edu

* Correspondence: cbalaji1988@gmail.com; Tel.: +91-948-994-9299

Received: 25 May 2018; Accepted: 13 July 2018; Published: 15 July 2018

Abstract: Attribute-based encryption (ABE) is used for achieving data confidentiality and access control in cloud environments. Most often ABE schemes are constructed using bilinear pairing which has a higher computational complexity, making algorithms inefficient to some extent. The motivation of this paper is on achieving user privacy during the interaction with attribute authorities by improving the efficiency of ABE schemes in terms of computational complexity. As a result the aim of this paper is two-fold; firstly, to propose an efficient Tate pairing algorithm based on multi-base number representation system using point halving (TP-MBNR-PH) with bases 1/2, 3, and 5 to reduce the cost of bilinear pairing operations and, secondly, the TP-MBNR-PH algorithm is applied in decentralized KP-ABE to compare its computational costs for encryption and decryption with existing schemes.

Keywords: attribute based encryption; TP-MBNR-PH; KP-ABE; multi-authority; cloud computing

1. Introduction

Cloud computing, as an emerging computing paradigm, empowers client to remotely store information on a cloud in order to access services on request. Over the past few years, it has been observed that cloud computing has become a full-fledged promising business idea for the IT sector. As data related to people and organizations resides in the cloud, to a large extent, a concern for security is addressed. This issue reduces the potentiality of cloud computing technologies in terms of giving protection and assurance to the end user information and, at the same time, it plaguies the market. In order to secure information from being disclosed, clients need to encipher their information before it is shared. Access control is elementary, as it is the primary line of defense that avoids unauthorized access to the shared information. In considering the above facts, attribute-based encryption (ABE) is given much more attention in providing information security and in comprehending fine-grained, one-to-numerous, and non-interactive access control. Thus it is evident that ABE supports both confidentiality and access control with a single encryption for data sharing in a cloud environment.

In 2005, Sahai and Waters [1] proposed another sort of IBE scheme called fuzzy IBE (FIBE) which compliments identities as a collection of descriptive attributes. FIBE is viewed as the primary idea of ABE in which the information owner encrypts a message to all users having a specific collection of attributes. In the same period, Nali et al. [2] also proposed a threshold-based ABE technique to convey the fact that this technique forestalls the collusion attacks and opens a new weakness in which threshold semantics are restricted in planning broader frameworks that require expressive access control. Data user, data owner, attribute authority (AA), and cloud storage server are the four kinds

of parties involved in ABE. In the ABE scheme, attributes are assumed to be the critical part. Attributes use public keys for encrypting data and are also utilized as an access policy for controlling users' access. It is realized in healthcare and smart grid applications that ABE provides fine-grained access control and broadcasting of a single encrypted message to a specific group of users, respectively. In view of the access policy, ensuing studies are generally ordered [3] either as a key-policy ABE (KP-ABE) or cipher text-policy ABE (CP-ABE).

In 2006, Goyal et al. [3] introduced the concept of KP-ABE in which each secret key is associated with an access structure that specifies the type of cipher text which can be decrypted by this secret key. The cipher texts are labelled with a collection of descriptive attributes. In case the attribute set fulfils the access structure indicated in the secret key, the user can decrypt the cipher text. It is one of the prominent encryption techniques with fine-grained access control for applications, say, sharing audit log information. The major drawback in this technique is that no sooner is the access policy built into the secret key, the data owner in this scheme cannot choose the person who is decrypting the cipher text, but can only decide a collection of attributes controlling the access of cipher texts. Later, Ostrovsky et al. [4] proposed a scheme with a non-monotonic access structure where the secret keys are stamped with a collection of attributes comprising positive and negative attributes. Analogously, the ABE scheme with a non-monotonic access structure elicits a more convoluted access policy. Unfortunately, the main flaw in this mechanism is that it doubles the size of the cipher text, and secret key and adds encryption/decryption overheads at the same time. Attrapadung et al. [5] suggested the first KP-ABE scheme with non-monotonic access structures and constant cipher text size. The drawback is that the secret key has a quadratic size in the number of attributes.

Goyal et al. [3] proposed the feasibility of a CP-ABE scheme, but not yet endeavored any constructions. In a CP-ABE scheme, a user's secret key is associated with a subjective number of attributes representing strings, and cipher text with an access structure. A user may have the capacity to decrypt a cipher text if user's attributes fulfil the access structure of the cipher text. In 2007, utilizing a monotonic access structure, Bethencourt et al. [6] proposed the main CP-ABE development. This technique sustains adaptable access control strategies like the KP-ABE [3] technique.

Considering the security aspects under the standard model, Cheung and Newport [7] contributed a provably secure CPABE scheme which, in turn, boosted the security proof in Bethencourt et al. [6]. This scheme supported AND gate on positive and negative attributes as its access policy and is proved to be the chosen plain text attack (CPA), secure under the decisional bilinear Diffie-Hellman (DBDH) assumption. Even though it has some advantages, there are some disadvantages, too. Mostly, this scheme is not adequately expressive because it supports only policies with logical conjunction. The next one is that the size of the cipher text and the secret key increments in a linear fashion with the aggregate number of attributes in this scheme. These two weaknesses made this scheme less proficient than Bethencourt et al.'s [6].

In view of Cheung and Newport's scheme [7], Nishide et al. [8] enhanced the effectiveness and accomplished hidden policies by proposing a scheme with multi-value attributes as its access policy. Emura et al. [9] utilized a similar access policy and proposed an enhanced scheme accomplishing a steady length of cipher text and a consistent number of bilinear pairing operations. Liang et al. [10] enhanced the bounded CP-ABE (BCP-ABE) by improving the proficiency of the encryption/decryption algorithm and reducing the length of the public key, secret key, and cipher text.

The initial ABE scheme was created utilizing single AA [1]. Later multiple-authority-based ABE (MA-ABE) was proposed in [11], since the single-authority ABE technique permitted a large volume of data at a single entity. In the MA-ABE technique, there are numerous AAs in charge of disjoint collections of attributes. In the customary MA-ABE technique, users co-operate with various AAs to obtain decryption credentials for their attributes. On the other hand, there is no security assurance for users; instead all AAs can share (collude) the specific user's data (attributes) to uncover the user's identity. Hence, the motivation of this paper is on achieving user privacy during the interaction with AAs by improving the efficiency of ABE schemes in terms of computational complexity. To the best of our knowledge, almost all the ABE schemes available are constructed from bilinear pairings.

However, bilinear pairing has a higher computational complexity, which makes algorithms inefficient to some extent. Therefore, the main focus of this paper is in reducing the cost of bilinear pairing operations to improve the efficiency of the ABE scheme.

1.1. Our Contributions

The main contributions of this paper are highlighted as follows:

- An efficient Tate pairing algorithm based on multi-base number representation system using point halving (TP-MBNR-PH) with bases 1/2, 3, and 5 has been proposed. This scheme mitigates the cost of bilinear pairing when compared to existing Tate pairing schemes. The efficiency is calculated using the computational costs and pre-computed costs of addition, subtraction, halving, tripling, and quintupling operations.
- The TP-MBNR-PH algorithm is applied in decentralized KP-ABE to show the reduction in computational costs for encryption and decryption when compared with existing schemes [12,13].

1.2. Paper Organization

The rest of this paper is organized as follows: Section 2 covers the related work. Section 3 deals with the proposed work of this paper. It consists of two subsections: firstly, Section 3.1 describes the proposed work of an efficient Tate pairing algorithm based on a multi-base number representation system using point halving (TP-MBNR-PH**)** with bases 1/2, 3, and 5; secondly, Section 3.2 describes the applicability of the TP-MBNR-PH algorithm in decentralized KP-ABE. Section 4 concludes the paper.

2. Related Work

There are two fundamental sorts of ABE, particularly cipher text-policy ABE (CP-ABE) and key-policy ABE (KP-ABE). The ABE scheme is categorized into two: single-authority ABE (SA-ABE) and multi-authority ABE (MA-ABE). In the MA-ABE scheme, there are two sub-categories; with a central authority (CA) and without a central authority. Chase introduced an MA-ABE scheme [14] utilizing a trusted CA for disbursing all the keys. The main drawback of utilizing a CA is that it increases the computation and communication cost. Lin et al. [15] resolved the secure threshold multi-authority fuzzy identity based encryption (threshold MA-FIBE) scheme in the absence of a central authority.

In the same lines, Chase and Chow in [11] introduced an MA-ABE scheme removing the CA using distributed pseudorandom functions. In this scheme, every pair of AAs firmly exchange a shared secret among them in the setup process. Users must submit their global identities (GIDs) to every AA to get the decryption credentials in [14]. This cleaves the user protection since a collection of perverted AAs can pool together each of the attributes that belong to the specific GID.

In [11], Chase and Chow introduced an anonymous key-issuing protocol to mitigate the privacy vulnerability in which a user can acquire the decryption keys from AAs without exposing his/her GID. Despite the fact that the scheme introduced by Chase and Chow avoids the central AA, all the AAs must be online and collude with each other to set up the ABE system. Thus, it is not fully decentralized. Furthermore, different protocols are proposed to decentralize the ABE scheme [11,14,16,17]; nonetheless, each scheme has its own benefits and bad marks.

The first known completely decentralized MA-ABE scheme is suggested in [16] where any party can turn into an AA and there is no prerequisite for any global co-ordination other than the production of a pioneer collection of common reference parameters. This overcomes the collusion vulnerability without providing co-ordination between AAs with novel strategies to tie key parts together and anticipate collusion attacks between users with various global identifiers. This scheme does not protect the user privacy as attributes of users are gathered by AAs following users' GIDs. The scheme in [11] considers privacy, however, it is not completely decentralized. Han et al. suggested a PP decentralized scheme for KP-ABE in [18] for preserving the user privacy based on the decisional bilinear Diffie-Hellman (DBDH) standard complexity assumption.

In [18], the GID of the user is utilized to tie all the decryption keys together, where blind key generation protocol has been used to issue the decryption keys. Subsequently, perverted AAs cannot pool the users' attributes by following the GIDs' of the users from the decryption keys. Unluckily, the scheme cannot counteract user collusion, thus, two users can pool their decryption keys to produce decryption keys for an unauthorized user [19]. This is because of weak binding between users' GID and the decryption keys.

Rahulamathavan et al. [12] constructed the privacy-preserving decentralized KP-ABE scheme in a cloud environment. It protects the users' privacy when they communicate with multiple authorities to obtain decryption credentials. It reduces the user collusion vulnerability found in [19] and used an anonymous key-issuing protocol based on anonymous credentials. Thus, it cannot generate decryption credentials for malicious users even if two or more users collude their keys. It is both leak-free and selective-failure blind. This scheme is verified using decisional bilinear Diffie-Hellman standard complexity assumption. Yang et al. [13] proposed a scheme to improve privacy and security in decentralizing multi-authority attribute-based encryption in cloud computing. Most often existing ABE schemes are constructed from bilinear pairings. This makes an algorithm inefficient due to its high computational complexity of bilinear pairing. In this paper, first, an efficient Tate pairing based on multi-base number representation system using point halving (TP-MBNR-PH) with point halving, tripling, and quintupling is proposed and then applied in decentralized KP-ABE [12] to determine its computational costs of encryption and decryption.

3. Proposed Work

The proposed work consists of two parts. Firstly, we propose an efficient Tate pairing algorithm based on multi-base number representation system using point halving (TP-MBNR-PH*)* with bases 1/2, 3, and 5 with the aim to reduce the cost of bilinear pairing operations. Secondly, the TP-MBNR-PH algorithm is applied in decentralized KP-ABE to determine its computational costs for encryption and decryption.

3.1. Proposed Tate Pairing Algorithm Construction

3.1.1. Bilinear Maps

Let G_1, G_2, and G_T be three cyclic groups of prime order q. G_1 and G_2 are a source group and G_T is a target group. Let g_1 and g_2 be generators of G_1 and G_2, respectively. A bilinear map e is defined as $e: G_1 \times G_2 \rightarrow G_T$ which has the following properties:

- Bilinearity: $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$, where $a, b \in Z$.
- Computability: The bilinear map e is efficiently computable by $G_1 \times G_2$ for any pairs.
- Non-degeneracy: $e(g_1, g_2) \neq 1$. This means all pairs of the source group do not map to the identity of the target group.

Note: If $G_1 = G_2$, then it is a symmetric map, otherwise it is an asymmetric map.

3.1.2. Point Halving (PH)

Fundamentally all the scalar multiplication is ascertained by utilizing the double and add method. However, Knuden (1999) and Schroeppel (2000), in parallel, proposed a strategy to speed up scalar multiplication on elliptic curves characterized over binary augmentation fields. Their technique depends on a novel elliptic curve primitive called point halving, which can be characterized as follows: Given a point Q of odd order, compute P such that $Q = 2P$. The point P is denoted as $1/2\ Q$. That means, in this technique the previous double and add method is replaced by the half and add method, which is the exact inverse operation of point doubling. The strategies replaced all point doublings in the double-and-add algorithm with another operation called point halving. This technique is executed for conducting scalar multiplication on non-super singular elliptic curves in characteristic 2. Point halving is applied to the curves with minimal two-torsion. Since, hypothetically, point halving is up toward three times as quick as point doubling, it is conceivable to

enhance the execution of scalar multiplication calculation $Q = nP$ by supplanting the double-and-add algorithm.

Let $P = (x, y)$ be a point on the elliptic curve defined over binary field using affine coordinates. A point doubling requires calculating the coordinates of the point $Q = 2P = (u, v)$ using the following equations:

$$\lambda = x + \frac{y}{x} \quad (1)$$

$$u = \lambda^2 + \lambda + a \quad (2)$$

$$v = x^2 + u(\lambda + 1) \quad (3)$$

Point halving is just the opposite, i.e., given $Q = (u, v)$, find $P = (x, y)$ such that $Q = 2P$. This is computed by solving Equation (2) for λ, Equation (3) for x, and finally, Equation (1) for y. This means that we have to solve $\lambda^2 + \lambda = u + a$, for λ, $v = x^2 + u(\lambda + 1)$ for x, and finally obtain $y = \lambda x + x^2$. A detailed analysis of the computational complexity of point halving was made in [20]. It was reported that the point halving method is 15% to 24% faster than point doubling.

3.1.3. The Double-Base Number System (DBNS)

In [21], a ternary/binary methodology was proposed for fast Elliptic Curve Cryptography. An equivalent tactic was suggested in [22] where an integer *k* is represented in the double-base number system. The following definitions are needed [23]:

Definition 1 (**S**-integer)**.** *Given a set of primes S, an S-integer is a positive integer whose prime factors all belong to S.*

Definition 2 (double-base number system)**.** *Given* p, q*, two relatively prime positive integers, the double-base number system (DBNS) is a representation scheme into which every positive integer n is represented as the sum or difference of* $\{p, q\}$*-integers, i.e., numbers of the form* $p^a q^b$*:* $n = \sum_{i=1}^{m} s_i\, p^{b_i} q^{t_i}$*, with* $s_i \in \{-1, 1\}$*, and* $b_i,\ t_i \geq 0$.

If the sequences of binary and ternary exponents decrease monotonically, i.e., $b_1 \geq b_2 \geq \ldots \geq b_m \geq 0$ and $t_1 \geq t_2 \geq \cdots \geq t_m \geq 0$, a double-base chain is formed.

Take the example of 314,159 as used in [24]. Its double-base chain representation is:

$$314{,}159 = 2^{12}3^4 - 2^{11}3^2 + 2^8 3^1 + 2^4 3^1 - 2^0 3^0$$

3.1.4. Multi-Base Number Representation (MBNR)

Let *k* be an integer and let $B = \{b_1, \ldots, b_l\}$ be a set of "small" integers. A representation of *k* as a sum of powers of elements of *B* is called a multi-base representation [25] of n using the base *B*. The base set size of the double-base representation, i.e., $|B| = 2$, and that of multi-base representation is greater than two, i.e., $|B| > 2$.

Definition: A multiple representation $l = \sum_{i=1}^{m} s_i\ 2^{b_i} 3^{t_i} 5^{r_i}$ using the bases {2, 3, 5} is called a step multi-base number representation, where each exponent $\{b_i\}$, $\{t_i\}$, and $\{r_i\}$ refes to separate monotonic decreasing sequences.

The MBNR is compared to DBNS, which is shorter in length and more redundant. For example, in Table 1, 200 has 3027 DBNS representation (base 2 and 3), 316,557 representations using the bases 2, 3 and 5 and has 4,827,147 representations using the bases 2, 3, 5, and 7.

Table 1. The number of MBNR of small numbers using various bases.

N	B = {2, 3}	B = {2, 3, 5}	B = {2, 3, 5, 7}
10	5	8	10
20	12	32	48
50	72	489	1266
100	402	8425	43,777
150	1296	63,446	586,862
200	3027	316,557	4,827,147
300	11,820	4,016,749	142,196,718

A multiple representation $l = \sum_{i=1}^{m} s_i \left(\frac{1}{2}\right)^{b_i} 3^{t_i} 5^{r_i}$ using the bases $\{\left(\frac{1}{2}\right), 3, 5\}$ is called a modified multi-base number representation [26], where each exponent $\{b_i\}$, $\{t_i\}$ and $\{r_i\}$ refers to separate monotonic decreasing sequences.

Take the example of 314,159 as used in [26]. Its MBNR is represented as:

$$314{,}159 = \left(\frac{1}{2}\right)^{17} 3^3 5^1 - \left(\frac{1}{2}\right)^{14} 3^3 5^1 + \left(\frac{1}{2}\right)^{10} 3^1 5^1 + \left(\frac{1}{2}\right)^{3} 3^1 5^0$$

The advantages of MBNR over DBNS are it is very shorter and more redundant. In the number of base elements, the number of representations of n grows aggressively. For example, 300 has 11,820 DBNS representations (base 2 and 3), 4,016,749 representations using the base 2, 3, and 5, and has 142,196,718 representations using the base 2, 3, 5, and 7.

In [26], mixed powers of 2, 3, and 5 have been proposed for representing the scalar. Instead, in [25], the authors proposed mixed powers of 1/2, 3, and 5 to obtain the faster elliptic curve cryptography (ECC) scalar multiplication. In this method, the point halving is used instead of point doubling and quadrupling while maintaining tripling and quintupling operations.

3.1.5. Proposed Tate Pairing Algorithm Based on Multi-Base Number Representation System Using Point Halving (TP-MBNR-PH)

We propose a Tate pairing algorithm based on multi-base number representation system using point halving.

The proposed Tate Pairing algorithm is based on Point Halving Technique. It takes input as an integer of MBNR representation with bases 1/2, 3, and 5 along with points *P* and *Q* which should be within the finite field FQ. Let L and V represented as line and vertical line passes through the points. N_1 be represented as function with the divisor. If the sign value s_1 is 1, then set N_1 to 1 as is shown in step 3, else N_1 is set to N_{-1} as is shown in step 6. The computation of N_{-1} is shown in step 2. The variables b_i, t_i, and r_i represents the exponents of base 1/2, 3, and 5, respectively, while inside the main for loop, TP-MBNR-PH initially calculates α, β, and γ which are the exponents of 1/2, 3, and 5 bases, as shown in steps 8–10. If the computed base 2 exponent α is equal to zero, then calculate the function N_1 as shown in step 13. If the computed base 3 exponent β is equal to zero, then compute the function N_1 as shown in step 17. If both of the computed bases α and β are equal to zero, then calculate N_1 as shown in step 21. If none of the above conditions are satisfied, then the algorithm computes N_1 as shown in steps 24, 26 and 28. In step 29, if the signed value s_{i+1} is equal to 1, then N_1 and C is computed as shown in step 30, else N_1 and C is computed as shown in step 32. TP-MBNR-PH finally returns $\mathrm{N}_1^{(q^k-1)/l}$.

Algorithm 1. TP-MBNR-PH

Input: An integer $l = \sum_{i=1}^{m} s_i \left(\frac{1}{2}\right)^{b_i} 3^{t_i} 5^{r_i}$, $s_i \in \{-1,1\}$, $b_1 \geq b_2 \geq \ldots \geq b_m \geq 0, t_1 \geq t_2 \geq \cdots \geq t_m \geq 0$ and $r_1 \geq r_2 \geq \cdots \geq r_m \geq 0$, $P = (x_P, y_P) \in E(F_q)[l]$, $Q = (x_Q, y_Q) \in E(F_{q^k})[l]$

Output: $e_l(P, Q)$

$C \leftarrow P$

$\mathrm{N}_{-1} \leftarrow \dfrac{1}{x_Q - x_P}$

If $s_1 = 1$, then

$\mathrm{N}_1 \leftarrow 1$

else

$\mathrm{N}_1 \leftarrow \mathrm{N}_{-1}$

for $\mathrm{i} = 1, 2, \ldots, \mathrm{n} - 1$ do

$\alpha \leftarrow b_i - b_{i+1}$

$\beta \leftarrow t_i - t_{i+1}$

$\gamma \leftarrow r_i - r_{i+1}$

If $\alpha = 0$ *then*

for $\mathrm{j} = 1, 2, \ldots\ldots, \beta$ do

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1^3 \frac{L_{C/4,C/4}(Q) L_{C/2,5C/2}(Q)}{V_{C/2}(Q) V_{3C}(Q)},$$

$C \leftarrow 3\mathrm{C}$

ElseIf $\beta = 0$ *then*

for $\mathrm{j} = 1, 2, \ldots\ldots, \alpha$ do

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1^2 \frac{L_{C/4,C/4}(Q)}{V_{C/2}(Q)}, C \leftarrow \frac{1}{2}\mathrm{C}$$

Else If $\alpha = 0$ *and* $\beta = 0$ *then*

for $\mathrm{j} = 1, 2, \ldots\ldots, \gamma$ do

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1^5 \frac{L_{C/4,C/4}^2(Q) L_{C/2,5C/2}(Q) L_{5C/2,5C/2}(Q)}{V_{C/2}^2(Q) V_{3C}(Q) V_{5C}(Q)},$$

$C \leftarrow 5\mathrm{C}$

Else

for $\mathrm{j} = 1, 2, \ldots, \alpha$ do

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1^2 \frac{L_{C/4,C/4}(Q)}{V_{C/2}(Q)}, C \leftarrow \frac{1}{2}\mathrm{C}$$

for $\mathrm{j} = 1, 2, \ldots, \beta$ do

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1^3 \frac{L_{C/4,C/4}(Q) L_{C/2,5C/2}(Q)}{V_{C/2}(Q) V_{3C}(Q)},$$

$C \leftarrow 3\mathrm{C}$

for $\mathrm{j} = 1, 2, \ldots, \gamma$ do

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1^5 \frac{L_{C/4,C/4}^2(Q) L_{C/2,5C/2}(Q) L_{5C/2,5C/2}(Q)}{V_{C/2}^2(Q) V_{3C}(Q) V_{5C}(Q)},$$

$C \leftarrow 5\mathrm{C}$

If $s_{i+1} = 1$ *then*

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1 \frac{L_{C,P}(Q)}{V_{C+P}(Q)}, C \leftarrow \mathrm{C} + \mathrm{P}$$

Else

$$\mathrm{N}_1 \leftarrow \mathrm{N}_1 . \mathrm{N}_{-1} \frac{L_{C,P}(Q)}{V_{C-P}(Q)}, C \leftarrow \mathrm{C} - \mathrm{P}$$

return $\mathrm{N}_1^{(q^k - 1)/l}$

3.1.6. Experimental Results

To obtain the results of the proposed TP-MBNR-PH, initially we have to apply the formula for computing the Tate pairing of elliptic curves over finite fields. Integers with at least 160 bit size which are represented with bases 1/2, 3, and 5 are used in Miller's algorithm. Table 2 shows the cost and pre-computed cost for the operations TADD, TSUB, THAL, TTRL, and TQNT in the proposed TP-MBNR-PH. Let TADD, TSUB, THAL, TTRL, and TQNT denote the addition, subtraction, halving, tripling, and quintupling operations, respectively, as shown in Table 2. Figure 1 and Table 3 shows the number of multiplication operation to compute Tate pairing using different methods. Let I, S and M denote the cost of inversion, squaring and multiplication in F_q^* respectively as shown in Table 1. Let I_k, S_k and $M_k(\approx k^{1.6}M)$ denote the cost of inversion, squaring, and multiplication in $F_{q^k}^*$, respectively, as shown in Table 1. Let $M_b(\approx kM)$ denote the cost of multiplication between F_q^* and $F_{q^k}^*$. An embedding degree denoted as k, which takes the values of 4, 6, and 8 [27]. In Table 4, we significantly improves the proposed TP-MBNR-PH and show the comparison of the proposed TP-MBNR-PH with an existing algorithm.

Table 2. Operational costs in the proposed Tate pairing algorithm.

Operation	Cost						Pre-Computed Cost				
	M_k	S_k	M_b	I	S	M	M_k	S_k	I_k	$M_{k/2}$	$I_{k/2}$
TADD	1	-	2.5	1	1	3	2	-	-	7	1
TSUB	1	-	-	1	1	2k+3	2	-	1	-	-
THAL	1	1	3.5	-	-	4	2	-	-	-	-
TTRL	3	1	2	1	4	9	-	1	-	-	-
TQNT	4	1	5	1	4	12	-	2	-	-	-

Table 3. Number of multiplication operations of proposed Tate pairing algorithm and existing algorithms.

Method	Embedding Degree		
	$k = 4$	$k = 6$	$k = 8$
Izu et al. [28]	12,328M	20,353M	28,379M
Kobayashi et al. [29]	9196M	13,685M	18,121M
Chang'an et al. [27]	8350M	12,554M	17,085M
Proposed Algorithm	6978.8M	10,805.8M	1,4642.8M

Table 4. Efficiency of proposed Tate pairing algorithm with the existing algorithms.

Method	Embedding Degree		
	$k = 4$	$k = 6$	$k = 8$
Izu et al. [28]	43.4%	46.9%	48.4%
Kobayashi et al. [29]	24.1%	21%	19.1%
Chang'an et al. [27]	16.4%	13.9%	14.3%

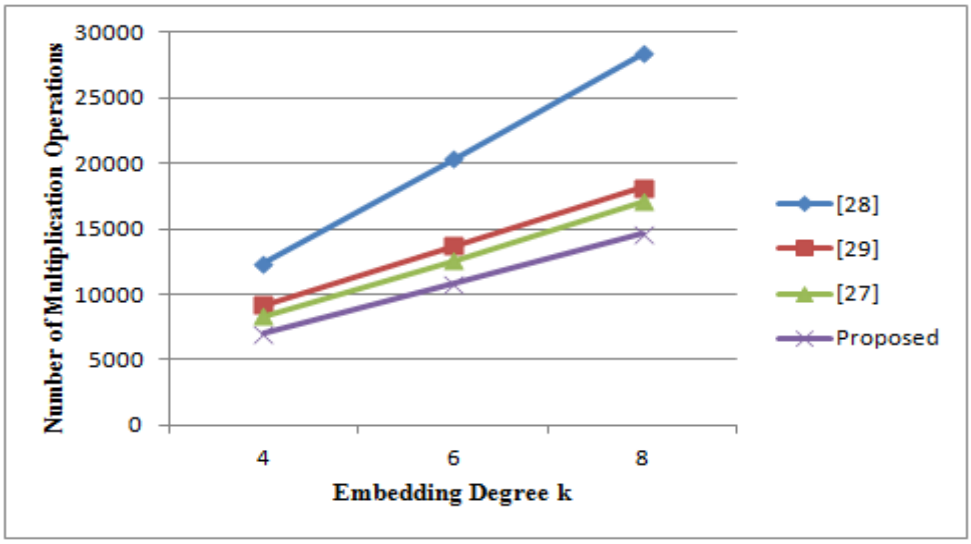

Figure 1. Comparison of number of multiplication operations based on the embedding degree.

3.1.7. Efficiency of the Proposed Algorithm

The total pre-computed cost of the proposed TP-MBNR-PH is:

$$T_{pre} = 6M_k + 3S_k + I_k + 7M_{k/2} + I_{k/2}$$

By taking $M_4 = 9M,\ M_6 = 18M,\ M_8 = 27M,\ M_k = kM,\ I = 10M,\ S = 0.8M,\ I_k = I + k^2M.$ The total cost of the proposed algorithm is:

$$T_{total} = b_{max}THAL + t_{max}TTRL + r_{max}TQTP + \frac{n}{2}(TADD + TSUB) + T_{pre}$$

$$T_{total} = (b_{max} + 3t_{max} + 4r_{max} + n + 6)\ M_k + (b_{max} + t_{max} + r_{max} + 3)\ S_k +$$

$$(\frac{7}{2}b_{max} + 2t_{max} + 5r_{max} + \frac{5}{4}n)\ M_b + (4b_{max} + 9t_{max} + 12r_{max} + (k+3)n)\ M$$

$$+(t_{max} + r_{max} + n)\ I\ + (4t_{max} + 4r_{max} + n)S + I_k\ + 7M_{k/2} + I_{k/2}$$

3.2. Applying the Proposed TP-MBNR-PH in a Decentralized KP-ABE Scheme

The TP-MBNR-PH algorithm is applied in a decentralized KP-ABE [12]. The detailed steps are as follows:

- *Global setup* (GS): Take input as a security parameter λ and it generates the bilinear group G_1 *and* $G_2\big(GS(1^\lambda) \to \{G_1, G_2\}\big)$ with prime order P. Let e: $G_1 \times G_1 \to G_2$ be the bilinear map and g_1, g_2, g_3 are generators of the group G_1. The N number of authorities are denoted as $\{A_1, A_2, \ldots, A_N\}$: A_k monitor n_k attributes i.e., $\tilde{A_k} = \{a_{k,1}, \ldots, a_{k,n_k}\}, \forall k$.
- *Attribute Authorities setup* (AAs): It is executed by each AA to randomly generate the Security parameter (SK_k) of authority A_k and public parameter (PK_k) of authority A_k:

$$\mathbb{Z}_p \xrightarrow{randomly} SK_k = \{\varkappa_k, \varrho_k, [\text{Ч}_{k,1}, \ldots, \text{Ч}_{k,n_k}]\}, \forall\ k$$

$$PK_k = \Big\{Y_k = e_{TP-MBNR-PH}(g_1, g_1)^{\varkappa_k}, Z_k = g_1^{\varrho_k}, [\text{Ч}_{k,1} = g_1^{\text{Ч}_{k,1}}, \ldots, \text{Ч}_{k,n_k} = g_1^{\text{Ч}_{k,n_k}}]\Big\}, \forall\ k$$

Each A_k specifies m_k as the minimum number of attributes required to satisfy the access structure $(m_k < n_k)$.

- *Key Generation* (KG): The attribute set of the user is $\tilde{A_u}: \tilde{A_u} \cap \tilde{A_k} = \tilde{A_u^k}, \forall\ k$. A_k generates $r_{k,u} \in_{randomly} \mathbb{Z}_p$ and polynomial q_x for each node x (including the leaves) $\mathbb{T}$. For each node x, the degree d_x of the polynomial q_x is $d_x = k_x - 1$ where k_x is the threshold value of that

node. For the root node $root$, set $q_{root}(0) = r_{k,u}$. For any other node x, $q_x(0) = q_{parent(x)}(index(x))$. Now decryption key for the user u is generates as follows:

$$DK = DK_{k,u} = g_1^{-\varkappa_k} g_2^{\frac{\varrho_k}{r_{k,u}+u}} g_3^{\frac{r_{k,u}}{\varrho_k+u}}, DK_{k,u}^1 = g_2^{\frac{1}{r_{k,u}+u}}, DK_{k,u}^j = g_3^{\frac{qa_{k,j}(0)}{(\varrho_k+u)\mathrm{Ч}_{k,j}}}, \forall a_{k,j} \in \tilde{A}_u^k$$

- $Encryption\ (E)$: Attribute set for the message m is $\tilde{A}_m$: $\tilde{A}_m \cap \tilde{A}_k = \tilde{A}_m^k, \forall k, i.e. \tilde{A}_m = \{\tilde{A}_m^1, \dots, \tilde{A}_m^k, \dots, \tilde{A}_m^N\}$. Data owner of message m randomly chooses $s \in_{\text{randomly}} \mathbb{Z}_p$, and output the ciphertext as follows:

$$C = \left\{ \begin{matrix} C_1 = m. \prod_{k \in I_C} e_{\text{TP-MBNR-PH}}(g_1, g_1)^{\varkappa_k s}, C_2 = g_1^s, C_3 = \prod_{k \in I_C} g_1^{\varrho_k s}, \\ \{C_{k,j} = \mathrm{Ч}_{k,j}^s\}_{\forall k \in I_C, a_{k,j} \in \tilde{A}_m^j} \end{matrix} \right\}$$

where I_C denotes the index set of the authorities.

- $Decryption\ (D)$: In order to decrypt C, the user u, computes $X, Y,$ and S_k as follows:

$$S_k = \prod_{a_{k,j} \in \tilde{A}_m^k} e_{\text{TP-MBNR-PH}}(C_{k,j}, DK_{k,u}^j)^{\Delta_{a_{k,j}, \tilde{A}_m^j}(0)}$$

$$Y = \prod_{k \in I_C} e_{\text{TP-MBNR-PH}}(C_3, DK_{k,u}^1)$$

$$S_k = \prod_{a_{k,j} \in \tilde{A}_m^k} e_{\text{TP-MBNR-PH}}(C_{k,j}, DK_{k,u}^j)^{\Delta_{a_{k,j}, \tilde{A}_m^j}(0)}$$

The user then decrypts the message m as follows:

$$m = \frac{C_1 X}{Y \prod_{k \in I_C} S_k}$$

The N number of AAs are denoted as $\{A_1, A_2, \dots, A_N\}$. Let $\tilde{A}_k = \{a_{k,1}, \dots, a_{k,n_k}\}$ be the attribute set managed by Attribute Authority (AA), which is denoted as A_k. The global setup algorithm takes the security parameter as input for generating bilinear group order G_1 and G_2. Each AA execute the AAs algorithm to randomly generate the public keys and the corresponding secret keys. The public-secret key pair for A_k is given as $\{SK_k = \{\varkappa_k, \varrho_k, [\mathrm{Ч}_{k,1}, \dots, \mathrm{Ч}_{k,n_k}]\}, \forall\ k, PK_k = \{Y_k, Z_k, [\mathrm{Ч}_{k,1}, \dots, \mathrm{Ч}_{k,n_k}]\}, \forall\ k\ \}$.

The key-generation algorithm issues the decryption keys to user u with a set of attributes, $\tilde{A}_u$. The output of the algorithm is a decryption key which permits the user to decrypt a message which is encrypted under a set of attributes $\tilde{A}_u^k$ which is based on the threshold policy, which relays on the tree-based access structure.

In the encryption algorithm, let $\tilde{A}_m$ denotes the attribute set which is used to encrypt message m, $\tilde{A}_m^k$ denotes the set of common attributes between message m and the AA, i.e., $\tilde{A}_m = \{\tilde{A}_m^1, \dots, \tilde{A}_m^k, \dots, \tilde{A}_m^N\}$. Additionally, let I_C denote the set of index of attribute authorities AAs involved in the ciphertext of message m. To encrypt the message$m$, the message owner has to generate s randomly and also he hast to calculate the cipher text C; $C = \{C_1, C_2, C_3, C_{k,j}, \forall a_{k,j} \in \tilde{A}_m^k\}$. To decrypt the message$m$, the user should have access to the decryption keys for the attributes. By executing the decryption algorithm, by following the four steps he can obtain the message m from the ciphertext as follows: (1) Initially the user has to compute X using C_2 and $DK_{k,u}$. (2) Next, the user uses decryption key $DK_{k,u}^1$ and C_3 to calculate Y. (3) Then the user has to use $DK_{k,u}^j$ and $C_{k,j}, a_{k,j} \in \tilde{A}_m^j$ and polynomial interpolation to obtain $r_{k,u}$. (4) Finally, the user can obtain the message m using C_1 and pre-computed values X, Y and S_k, $\forall k$.

3.2.1. Anonymous Key-Issuing Protocol

In order to avoid user collusions, we have used anonymous key-issuing protocol which is based on anonymous credential system which, in turn, allows users to access decryption keys from the AAs without enlightening their GIDs. The user U and the attribute authority A_k jointly construct the key-issuing protocol, which consists of the following steps:

- The two-party protocol (2PC) is used for the interaction between the user u and the attribute authority A_k. The 2PC protocol takes $u, \mathfrak{H}_1$ and $\mathfrak{H}_2$ from user $\{r_{k,u}, \varrho_k\}$ from A_k and return $x = (u + \varrho_k)\mathfrak{H}_1 \bmod p$ and $y = (u + r_{k,u})\mathfrak{H}_2 \bmod p$ to A_k.
- Once the 2PC protocol gets executed, the user u now computes $P = g_1^{\frac{1}{\mathfrak{H}_1\mathfrak{H}_2}}, Q = g_2^{\frac{1}{\mathfrak{H}_2}}$ and $R = g_3^{\frac{1}{\mathfrak{H}_1}}$ and then sends to A_k.
- Attribute Authority A_k computes $\widetilde{D_{k,u}}$, $\widetilde{D_{k,u}^1}$, $\widetilde{D_{k,u}^J}$, $\forall a_{k,j} \in \tilde{A}_u^k$ and proof of knowledge with the help of P, Q, R, x, and y and send them to the user:

$$\widetilde{D_{k,u}} = P^{-\varkappa_k} Q^{\frac{\varrho_k}{x}} R^{\frac{r_{k,u}}{y}}$$

$$\widetilde{D_{k,u}^1} = Q^{\frac{1}{x}}$$

$$\widetilde{D_{k,u}^J} = R^{\frac{r_{k,u}(a_{k,j})}{y\text{ч}_{k,j}}}, \forall a_{k,j} \in \tilde{A}_u^k$$

- User u exponentiates the received values by $\mathfrak{H}_1\mathfrak{H}_2$ to get the decryption keys.

The key advantage of the proposed key issuing protocol is both $leak\ free$ and $selective-failure\ blind$. Suppose for the message m, the attribute set is $\tilde{A}_m = \{a_1, a_2\}$. If the users U_1 and U_2 with identifiers u_1 and u_2 respectively have access to decryption credential for attribute $\{a_1\}$, while another user U_3 with identifier u_3 has access to attribute $\{a_2\}$ alone; none of the users can decrypt the ciphertext alone. However, there is a possibility that users can collude together so that they can generate the decryption credentials to decrypt the cipher text. This algorithm overcomes the user collusion vulnerability since u_1 and u_2 cannot be substituted with u_3 without the knowledge of ϱ_1, r_{1,u_1}, and r_{1,u_2}.

3.2.2. Proposed Scheme: Proof of Security

Decisional Bilinear Diffie-Hellman (DBDH) assumption: Let $a, b, c, z \in \mathbb{Z}_p$ be chosen at random, G be the group of prime order q and g is the generator of the group G. The DBDH problem [30] is a problem that no polynomial time adversary is able to distinguish the tuple $(g^a, g^b, g^c, e(g,g)^{abc})$ from the tuple $(g^a, g^b, g^c, e(g,g)^z)$ with a non-negligible advantage. This can be formalized as follows:

$$\left(\left|\Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g,g)^{abc}) = 0] - \Pr\left[\mathcal{A}\left(g, g^a, g^b, g^c, e(g,g)^z\right) = 0\right]\right| \geq \varepsilon\right)$$

Theorem 1. *Under Decisional Bilinear Diffie-Hellman (DBDH) assumptions, no polynomial time attacker can selectively break the proposed system.*

Proof. The security game is based on the hardness of the DBDH assumption. Suppose attacker atk can win the FH-CP-ABE game with advantage ε. We construct a simulator sim that can distinguish a DBDH tuple from a random tuple with advantage $\frac{\varepsilon}{2}$. Let G_1 be the source group and G_2 be the target group. Let g be the generator of the group G_1. The challenger chooses the fair binary coin $\mathfrak{h} \in \{0,1\}$, $g \in G_1$, $R \in G_2$ and $a, b, c \in \mathbb{Z}_p$. If $\mathfrak{h} = 0$, then the challenger defines T to be $e(g,g)^{abc}$. Otherwise, he sets $T = e(g,g)^z$ *or* R. The challenger then gives the simulator the DBDH details and then simulator sim now plays the role of challenger in the security game.

Init: During the init phase, sim receives the challenge access structure $\mathcal{A}^*$ from attacker atk.

Setup: To provide a public key PK to atk, sim randomly chooses $\varkappa_k' \in \mathbb{Z}_p$ and note $\varkappa_k = \varkappa_k' + ab$. It calculates $e(g,g)^{\varkappa_k}$ as: $e(g,g)^{\varkappa_k} = e(g,g)^{\varkappa_k'}.e(g,g)^{ab}$. Finally, sim sends public key $PK$to atk.

Phase 1: During this phase, atk submits an attribute set $\mathcal{W}_j \in \mathcal{A}$ such that $\mathcal{W}_j \notin \mathcal{A}^*$, to sim. Simulator simchooses $r_{k,u}' \in_{randomly} \mathbb{Z}_p$. It can be obtained as follows: $DK_{k,u} = g^{-\varkappa_k} g^{\frac{(\gamma+\delta)\varrho_k}{r_{k,u}'+u}} g^{\frac{\gamma\eta r_{k,u}'}{\varrho_k+u}}$. For each attribute in $\mathcal{W}_j$, sim has to choose $\mathrm{ч}_{k,j} \in_{randomly} \mathbb{Z}_p$. It computes the rest of the secret key as follows: $DK_{k,u}^1 = g^{\frac{(\gamma+\delta)}{r_{k,u}'+u}}$, $DK_{k,u}^j = g^{\frac{\gamma\eta q_{parent(a_{k,j})}(index(a_{k,j}))}{(\varrho_k+u)\mathrm{ч}_{k,j}}}$. Finally, sim sends the SK to atk.

Challenge: The attacker atk submits two equal length messages m_1 and m_2 along with a challenge access structure $\mathcal{A}^*$. simrandomly generates a bit $\hat{\mathrm{n}} \in \{0,1\}$ and computes CT^* as $\mathrm{C}_1 = \frac{m_{\hat{\mathrm{n}}} Y \prod_{k \in I_C} S_k}{X}$. Finally, sim sends the CT^* to atk.

Phase 2: Same as Phase 1.

Guess: The attacker atk outputs a guess $\hat{\mathrm{n}}'$of $\hat{\mathrm{n}}$. If $\hat{\mathrm{n}} = \hat{\mathrm{n}}'$, simulator sim guesses that $T = e(g,g)^{abc}$. Otherwise, T is a random target group element in G_2.

The advantage of the attacker is ε, when $T = e(g,g)^{abc}$. The advantage of the attacker is $\frac{1}{2}$, when T is a random target group element in G_2. Finally, the advantage of the simulator in this security game is $\frac{\varepsilon}{2}$.

3.2.3. Experimental Results

In this section, we show the total computation cost of encryption and decryption for the MA-ABE. The proposed Tate pairing algorithm is discussed in detail in Section 3. The proposed decentralized KP-ABE is constructed with the help of the proposed TP-MBNR-PH and compared with MA-ABE using an existing Tate pairing algorithm. In this experiment, we used an Intel Core i3-3217U CPU processor (Intel, China) with 1.80 GHz and 8 GB RAM. Let us assume the number of attribute authorities, $N = 2$, and say, attribute n varies from 10 to 50.

Figure 2 depicts the comparison of total encryption cost of MA-ABE for the proposed scheme and [12,13]. The time complexity of proposed encryption algorithm increases linearly with respect to the attributes. Figure 2 clearly shows the significant improvement of the proposed encryption algorithm when compared with the existing schemes [12,13]. Figure 3 depicts the comparison of the total decryption cost of MA-ABE for the proposed scheme and [12,13]. The time complexity of the proposed decryption algorithm increases linearly with respect to the attributes. Figure 3 clearly shows the significant improvement of the proposed decryption algorithm when compared with the existing schemes [12,13].

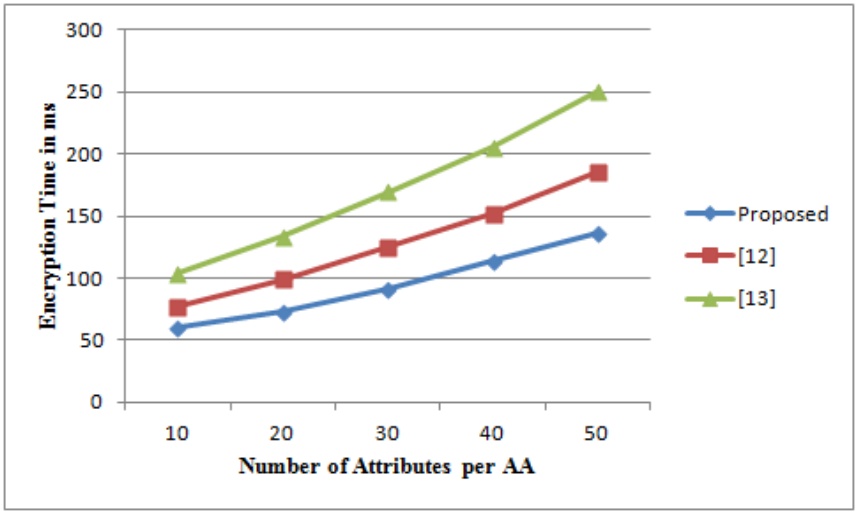

Figure 2. Comparison of the computation cost for encryption based on the number of attributes.

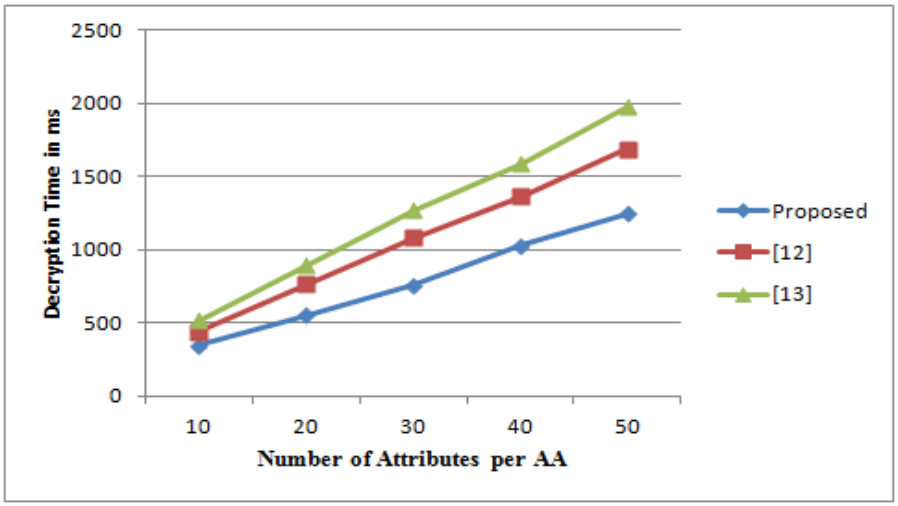

Figure 3. Comparison of the computation cost for decryption based on the number of attributes.

4. Conclusions

In this paper, we presented an efficient Tate pairing algorithm based on multi-base number representation system using point halving (TP-MBNR-PH) with bases 1/2, 3, and 5 to reduce the cost of bilinear pairing operations. The efficiency of the proposed algorithm has been significantly improved when compared with the existing Tate pairing algorithms. In [12,13], the schemes have proved that an anonymous key issuing protocol is free from leaks, selective-failures, and avoids user collusion. The TP-MBNR-PH algorithm is then applied in decentralized KP-ABE [12] in cloud environment to compute the cost for encryption and decryption. It is inferred that the TP-MBNR-PH algorithm, when applied in a KP-ABE scheme, has shown a significant improvement than the existing schemes [12,13] in terms of computational cost for encryption and decryption.

Author Contributions: B.C. developed the idea, performed the experiments and also wrote the paper, R.B. supervised the research.

Funding: This research work is funded by the Ministry of Human Resource Development (MHRD) under the Government of India.

Acknowledgments: This research work is supported and funded by the Ministry of Human Resource Development (MHRD) under the Government of India.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sahai, A.; Waters, B. Fuzzy identity-based encryption. In *Advances in Cryptology—EUROCRYPT 2005*; Lecture Notes in Computer Science; Springer: Berlin, Germany, 2005; Volume 3494, pp. 457–473.
2. Nali, D.; Adams, C.; Miri, A. Using threshold attribute based encryption for practical biometric-based access control. *Int. J. Netw. Secur.* **2005**, *1*, 173–182.
3. Goyal, V.; Pandey, O.; Sahai, A.; Waters, B. Attribute based encryption for fine-grained access control of encrypted data. In Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06), Alexandria, VA, USA, 30 October–3 November 2006; pp. 89–98.
4. Ostrovsky, R.; Sahai, A.; Waters, B. Attribute-based encryption with non-monotonic access structures. In Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07), Alexandria, VA, USA, 29 October–2 November 2007; pp. 195–203.
5. Attrapadung, N.; Libert, B.; de Panafieu, E. Expressive key policy attribute-based encryption with constant-size ciphertexts. In *Public Key Cryptography—PKC 2011*; Springer: Berlin, Germany, 2011; Volume 6571, pp. 90–108.

6. Bethencourt, J.; Sahai, A.; Waters, B. Ciphertext-policy attribute-based encryption. In Proceedings of the IEEE Symposium on Security and Privacy (SP '07), Oakland, CA, USA, 20–23 May 2007; pp. 321–334.
7. Cheung, L.; Newport, C. Provably secure ciphertext policy ABE. In Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07), Alexandria, VA, USA, 29 October–2 November 2007; pp. 456–465.
8. Nishide, T.; Yoneyama, K.; Ohta, K. Attribute-based encryption with partially hidden encryptor-specified access structures. In *Applied Cryptography and Network Security (ACNS 2008)*; Springer: Berlin, Germany, 2008; pp. 111–129.
9. Emura, K.; Miyaji, A.; Omote, K.; Nomura, A.; Soshi, M. A ciphertext-policy attribute-based encryption scheme with constant ciphertext length. *Int. J. Appl. Cryptogr.* **2010**, *2*, 46–59.
10. Liang, X.; Cao, Z.; Lin, H.; Xing, D. Provably secure and efficient bounded ciphertext policy attribute based encryption. In Proceedings of the 4th International Symposium on ACM Symposium on Information, Computer and Communications Security (ASIACCS '09), Sydney, Australia, 10–12 March 2009; pp. 343–352.
11. Chase, M.; Chow, S.S. Improving privacy and security in multi-authority attribute-based encryption. In Proceedings of the 16th ACM Conference on Computer and Communications Security, New York, NY, USA, 9–13 November 2009; pp. 121–130.
12. Rahulamathavan, Y.; Veluru, S.; Han, J.; Li, F.; Rajarajan, M.; Lu, R. User Collusion Avoidance Scheme for Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption. IEEE Trans. Comput. 2016, 65, 2939–2946.
13. Yang, Y.; Chen, X.; Chen, H.; Du, X. Improving Privacy and Security in Decentralizing Multi-Authority Attribute-Based Encryption in Cloud Computing. IEEE Access 2018, 6, 18009–18021.
14. Chase, M. Multi-authority attribute based encryption. In *Theory of Cryptography*; Springer: Berlin, Germany, 2007; pp. 515–534.
15. Lin, H.; Cao, Z.; Liang, X.; Shao, J. Secure threshold multi authority attribute based encryption without a central authority. *Inf. Sci.* **2010**, *180*, 2618–2632.
16. Lewko, A.; Waters, B. Decentralizing attribute-based encryption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 2011, pp. 568–588.
17. Waters, B. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In *International Workshop on Public Key Cryptography*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 53–70.
18. Han, J.; Susilo, W.; Mu, Y.; Yan, J. Privacy-preserving decentralized key-policy attribute-based encryption. *IEEE Trans. Parallel Distrib. Syst*. **2012**, *23*, 2150–2162.
19. Ge, A.; Zhang, J.; Zhang, R.; Ma, C.; Zhang, Z. Security analysis of a privacy—preserving decentralized key-policy attribute-based encryption scheme. *IEEE Trans. Parallel Distrib. Syst*. **2013**, *24*, 2319–2321.
20. Fong, K.; Hankerson, D.; Lopez, J.; Menezes, A. Field Inversion and Point Halving Revisited. *IEEE Trans. Comput.* **2004**, *53*, 1047–1059.
21. Ciet, M.; Joye, M.; Lauter, K.; Montgomery, P.L. Trading Inversions for Multiplications in Elliptic Curve Cryptography. *Des. Codes Cryptogr*. **2006**, *39*, 189–206.
22. Dimitrov, V.S.; Imbert, L.; Mishra, P.K. Fast Elliptic Curve Point Multiplication using Double-Base Chains. *Cryptol. Epr. Arch.* **2005**, *2005*, 69.
23. Dimitrov, V.; Imbert, L.; Mishra, P. The double-base number system and its application to elliptic curve cryptography. *Math. Comput.* **2008**, *77*, 1075–1104.
24. Wong, K.W.; Edward, C.W.; Lee, L.M.; Liao, X. Fast elliptic scalar multiplication using new double-base chain and point halving. *Appl. Math. Comput.* **2006**, *183*, 1000–1007.
25. Mishra, P.K.; Dimitrov, V. Efficient quintuple formulas for elliptic curves and efficient scalar multiplication using multibase number representation. In *International Conference on Information Security*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 390–406.

26. Ismail, A.M.; Said, M.R.M.; Atan, K.M.; Rakhimov, I.S. An Algorithm to enhance Elliptic Curves scalar Multiplication Combining MBNR with point halving. *Appl. Math. Sci.* **2010**, *4*, 259–261.
27. Zhao, C.; Zhang, F.; Huang, J. Efficient Tate Pairing Computation Using Double-Base Chains. *Sci. China Ser. F Inf. Sci.* **2008**, *51*, 1096–1105.
28. Izu, T.; Takagi, T. Efficient computations of the Tate pairing for the large MOV degrees. In *International Conference on Information Security and Cryptology*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 283–297.
29. Kobayashi, T.; Aoki, K.; Imai, H. Efficient algorithms for Tate pairing. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2006**, *1*, 134–143.
30. Chandrasekaran, B.; Balakrishnan, R.; Nogami, Y. Secure Data Communication using File Hierarchy Attribute Based Encryption in Wireless Body Area Networks. *J. Commun. Softw. Syst.* **2018**, *14*, 75–81.