An Expressive, Lightweight and Secure Construction of Key Policy Attribute-Based Cloud Data Sharing Access Control

Guofen Lin, Hanshu Hong, Yunhao Xia and Zhixin Sun

The Key Lab of Broadband Wireless Communication and Sensor Network Technology, Nanjing University of Posts and Telecommunications, Nanjing China
Email: 2015070249@njupt.edu.cn, 2014070244@njupt.edu.cn, 2015070108@njupt.edu.cn, sunzx@njupt.edu.cn

Abstract. Attribute-based encryption (ABE) is an interesting cryptographic technique for flexible cloud data sharing access control. However, some open challenges hinder its practical application. In previous schemes, all attributes are considered as in the same status while they are not in most of practical scenarios. Meanwhile, the size of access policy increases dramatically with the raise of its expressiveness complexity. In addition, current research hardly notices that mobile front-end devices, such as smartphones, are poor in computational performance while too much bilinear pairing computation is needed for ABE. In this paper, we propose a key-policy weighted attribute-based encryption without bilinear pairing computation (KP-WABE-WB) for secure cloud data sharing access control. A simple weighted mechanism is presented to describe different importance of each attribute. We introduce a novel construction of ABE without executing any bilinear pairing computation. Compared to previous schemes, our scheme has a better performance in expressiveness of access policy and computational efficiency.

1. Introduction

With improvements in computational technology and large-scale networks, sharing data with others becomes correspondingly more convenient. Additionally, digital resources are more easily obtained via cloud computing and storage. Since cloud data sharing requires off-premises infrastructure that some organizations jointly held, remote storage are somehow threatening privacy of data owners. Therefore, enforcing the protection of personal, confidential and sensitive data stored in the cloud is extremely crucial. The simultaneous participation of large numbers of users requires fine-grained access control for data sharing. Attribute-based encryption (ABE) is a promising cryptographic primitive that offers an interesting solution to secure and flexible data sharing. ABE has an inherent one-to-many property, which means a single key can decrypt different ciphertexts or different keys can decrypt the same ciphertext. There are two types of ABE, called ciphertext policy ABE (CP-ABE) and key policy ABE (KP-ABE). For CP-ABE, the access policy is embedded into a ciphertext and the attribute set is embedded into a private key. For KP-ABE, the access policy is embedded into a private key and the attribute set is embedded into a ciphertext.

However, there are still a lot of open challenges concerning the practical realization of ABEs especially with respect to the limited expressiveness of access policy and heavy computational overhead. In some ABE schemes, the access policy is built based on threshold gate. When the distance between the attribute set and the access policy is shorter than a given threshold, the data receiver can extract the plaintext correctly. For most of current ABE schemes, they adopt access tree as their access policy, which consists of a plenty of “AND” gate, “OR” gate. Although the access tree enriches the form of ABE’s access policy, we are still not satisfied with its expressiveness. For one thing, all
attributes are considered as in the same status while they are not in most of practical scenarios. For another thing, the size of access policy increases dramatically with the raise of its expressiveness complexity. Together with the growth of mobile applications, mobile cloud services have been introduced as a potential trend in cloud computing. Current research hardly notices that mobile front-end devices, such as smartphones, are poor in computation performance while too much bilinear pairing computation is need for ABE. As a result, the run time may be horribly unacceptable.

1.1. Related Work
In 2005, Sahai and Waters [1] proposed a fuzzy identity-based encryption (FIBE) based on classic identity-based encryption. Since FIBE indicated some many key features of ABE, it laid a theoretical foundation of subsequent research into ABE. In 2006, a formal definition of ABE [2] was proposed, which indicated that its construction based on FIBE is a key policy ABE (KP-ABE). That means each private key is associated with an access policy and each ciphertext is associated with a set of attributes. Inspired by reference, Bethencourt, Sahai and Waters [3] proposed a concrete construction of CP-ABE, in which a data owner can flexibly define the access policy before data is encrypted. However, their construction is more complicated than KP-ABE. Cheung and Newport [4] proposed a novel access tree contain only AND gates with positive and negative attributes to enhance expressiveness of ABE access policy. Waters [5] proposed a high efficient CP-ABE that first makes ciphertext size, encryption time and decryption time increase linearly with the complexity of access policy. In addition, the proposed scheme is proven to be selectively secure with the reduction to Decisional Parallel Bilinear Diffie-Hellman Exponent Assumption (DPBDHE). Green, Hohenberger and Waters [6] proposed a concept of outsourcing decryption ABE (OD-ABE) that delegates a high-performance server to execute part of the decryption while leaving no information with it. Finally, only a small amount of computation is needed for the data receiver to extract a plaintext. In the original OD-ABE, however, the untrusted proxy server risks the confidentiality of ciphertext and transform keys. Furthermore, wrong computation may result in unavailability of the whole system. To solve this problem, an attribute-based encryption with verifiable outsourced decryption (VOD-ABE) [7] was proposed, which demonstrated the necessity of outsourcing verifiability for OD-ABE. Lin, Zhang, Ma and Wang [8] proposed an improved VOD-ABE based on key encapsulation mechanism (KEM). Their experiment demonstrated that ciphertext size, cost of encryption and decryption are nearly half of those in [7]. Based on a concept of hierarchical generalized attributes, a hierarchical multi-authority mechanism for CP-ABE was proposed in [9]. When a user defines an access structure and requests data encryption, each key generation center (KGA) generates corresponding access policy and private key of its level so that security of key management is guaranteed even there exist a compromised KGA engaged in key issuing. In [10], a proxy re-encryption scheme for multi-authority ABE was proposed to realize efficient and fine-grained revocation. In this scheme, access policies are mapped to a weighted access list, which reduces computation overhead during generating and issuing private keys. Hur [11] proposed a secure and efficient attribute-based data sharing system. Without adding any infrastructure, they provided a novel solution to key escrow problem in single-authority system by introducing two-party computation (2PC) between key generation center and data storing center. Ouahla and Nguyen [12] considered the limitation of energy and computation of nodes in Internet of Things, and proposed a novel CP-ABE by introducing a pre-computation technique, which computes and stores some essential elements before encryption occurs. Qin et al. [13] indicated that although KP-ABE does not allow data owners to define access policies, it helps to build an efficient access control for cloud data storage. Based on this, they proposed a redefinable KP-ABE, which provides flexible access control but also markedly reduces ciphertext size.

As mentioned above, most of current ABE schemes adopt threshold gate or access tree to build their access policy. In these cases, all attributes are considered as in the same status while they are not in most of practical scenarios. Meanwhile, the size of access policy increases dramatically with the raise of its expressiveness complexity. Moreover, current research hardly notices that mobile front-end devices, such as smartphones, are poor in computational performance while heavy computation overhead especially too much bilinear pairing computation is taken over by them. Thus, we are not satisfied with current ABEs in terms of the limited expressiveness of access policy and heavy
computational overhead.

1.2. Our Contributions
In this work, we propose a key policy weighted attribute-based encryption without bilinear pairing computation (KP-WABE-WB) aiming to enhance expressiveness and computational efficiency of secure cloud data sharing access control. The main contributions are summarized as follows:

- A simple weighted mechanism is presented to describe different importance of each attribute. As a result, the weighted mechanism not only enhances the expressiveness of access policy but also simplifies its structure.
- We introduce a novel attribute-based encryption without executing any bilinear pairing computation. Therefore, the computational efficiency is markedly improved.
- We propose a reduction of the proposed scheme to Computational Diffie-Hellman (CDH) Assumption. With the help of this reduction, we prove that our KP-WABE-WB is secure under chosen-ciphertext attacks in the random oracle model.

1.3. Organization
The rest of paper is organized as follows: We give essential preliminaries in Section II, propose main construction of KP-WABE-WB in Section III, provide comprehensive analysis in Section IV, and draw conclusion and highlight our future work in Section V.

2. Preliminaries

2.1 Complexity Assumption
Our construction is based on the following complexity assumption:

**Computational Diffie-Hellman (CDH) Assumption:** Given an additive cyclic group $\mathbb{G}$ with prime order $q$. Let $g$ be a generator of $\mathbb{G}$. There are two random elements $a, b \in \mathbb{Z}_q^*$. Given $P_1 = aP$ and $P_2 = bP$, no probabilistic polynomial-time algorithm can compute $P_3 = abP$.

2.2 Modeling Syntax
The model of our construction is shown in Figure 1. As depicted, there are four entities involved in cloud data sharing, which is the key authority, the data owner, the cloud storage center, and the data receiver. The key authority is responsible for generating and issuing keys. The data owner possesses data to be shared via cloud service. The cloud storage center is responsible for storing and managing encrypted data.

**Figure 1** The model of KP-WABE-WB.

The data receiver wants to obtain data stored in the cloud storage center. Precisely, the KP-WABE-WB consists of the following 4 algorithms:

- **Setup:** The setup algorithm is run by the key authority. It takes as input a security parameter $k \in \mathbb{Z}^+$, then it outputs the master secret $MK$ and the public parameter $PK$.
Noting that the master secret is kept by the key authority in secret while the public parameter is available for all entities.

- \textit{KeyGen}(MK, PK, \Lambda) \rightarrow SK : The key generation algorithm is run by the key authority when receiving a key generation query from a data receiver. It takes as input the master secret, the public parameter, and an access policy. Then, it outputs a private key \( SK \) associated with this access policy.

- \textit{Encrypt}(PK, S, M) \rightarrow CT : The encryption algorithm is run by a data receiver. It takes as input the public parameter \( PK \), an access policy \( \Lambda \), and a plaintext \( M \). Finally, it returns a ciphertext \( CT \) associated with the attribute set \( S \).

- \textit{Decrypt}(SK, CT) \rightarrow M : The decryption algorithm takes as input the private key of a data receiver and a ciphertext. If the attribute set of a ciphertext satisfies the access policy of a private key, it outputs the plaintext correctly. Otherwise, it aborts the decryption.

3. Main Construction

3.1 The Setup Algorithm

Let \( \Omega = (\theta_1, \theta_2, \ldots, \theta_n) \) be the global attribute space and \( w = \{w_1, w_2, \ldots, w_n\} \) be the collection of attributes' weight. Given a security parameter \( k \in \mathbb{Z}^+ \), the setup algorithm chooses a cyclic group \( \mathbb{G} \) with prime order \( q \). Let \( P \) be a generator of \( \mathbb{G} \). For arbitrary \( \theta_i \in \Omega \), it selects a unique random element \( t_i \in \mathbb{Z}_q^+ \) and compute \( T_i = t_iP \). Choose \( y \in \mathbb{Z}_q^+ \) and compute \( Y = yP \). Define a hash function \( H : \mathbb{Z}_q^+ \rightarrow \mathbb{G} \). Then it sets the master secret:

\[
MK = \langle \{t_i\}, y \rangle
\]

And outputs the public parameter as follows:

\[
PK = \langle q, \mathbb{G}, P, \Omega, T_i, \{y\}, H \rangle
\]

3.2 The Key Generation Algorithm

It takes the public parameter \( PK \), the master secret \( MK \) and an access policy \( \Lambda \) as input. It selects a threshold gate \( (k, n) \), in which \( n = \sum_{\theta_i \in \Lambda} w_i \) and \( k \) is selected based on the access policy that satisfies \( 1 \leq k \leq n \). For arbitrary attribute in access policy, it allocates \( |w_i| \) index numbers:

\[
\{\forall \theta_i \in \Lambda : index(x_1), index(x_2), \ldots, index(x_{|w_i|})\}
\]

The algorithm selects a random element \( r \in \mathbb{Z}_q^+ \) and generates a random polynomial \( q_{\text{rest}} \) with degree \( d = k - 1 \), which satisfies \( q_{\text{rest}}(0) = ry \). For each index number, set the following polynomial:

\[
q_{\text{rest}}(index(x, i)) = q_{\text{rest}}(0)
\]

Define \( D = r \) and \( D_i = \sum_{\theta_i \in S} q_{\text{rest}}(0) + t_i \). Finally, it outputs the private key:

\[
SK = \langle D, \{D_i\} \rangle
\]

3.3 The Encryption Algorithm

The encryption algorithm takes as input the public parameter \( PK \), an attribute set \( S \) and a plaintext \( M \). It first selects a random element \( s \in \mathbb{Z}_q^+ \) and computes \( C_i = sP \) and \( C = H(sY) \oplus M \). For each attribute \( \theta_j \in S \), it computes \( C_j = sT \). Finally, it outputs the ciphertext as follows:
3.4 The Decryption Algorithm

The decryption algorithm takes as input a private key $SK$ and a ciphertext $CT$. Let $x$ be a leaf node in an access policy $\land$ that represents the attribute $\theta_i$. For each $x \in \land$, define a Lagrange coefficient as follows:

$$\Delta_{\omega_i,\land}(x) = \prod_{j \in \omega_i\setminus\{x\}} \frac{x - j}{idx - j}$$

(7)

Where $idx = index(x,i)$ and $\land' = \{x \in \land : index(x,i)\}$. If $\theta_i \in S$, it computes:

$$DecrpytNode(C_i, C_i', D_i) = (D_i \cdot C_i - C_i) \cdot \Delta_{\omega_i,\land}(0)$$

$$= ((\sum_{j=1}^{m} q_{\theta_i}(0) + t_j) \cdot sP - sT_i) \cdot \Delta_{\omega_i,\land}(0)$$

$$= \sum_{j=1}^{m} q_{\theta_i}(0) \cdot \Delta_{\omega_i,\land}(0) \cdot sP$$

(8)

Otherwise, the algorithm returns $\perp$ to indicate error. If the attribute set satisfies the access policy, it extracts the plaintext as follows:

$$M = H(D^{-1} \sum_{\theta_i \in \land'} DecrpytNode(C_i, C_i', D_i)) \oplus C$$

(9)

3.5 Correctness Proof

In the decryption algorithm, for any authorized data receiver who launches a decryption query, if attribute set of a ciphertext satisfies the access policy of its private key, it can recover $sY$ as follows:

$$D^{-1} \sum_{\theta_i \in \land'} DecrpytNode(C_i, C_i', D_i)$$

$$= r^{-1} \sum_{\theta_i \in \land'} \sum_{j=1}^{m} q_{\theta_i}(0) \cdot \Delta_{\omega_i,\land}(0) \cdot sP$$

$$= r^{-1} \sum_{\theta_i \in \land'} q_{\theta_i}(index(x,i)) \cdot \Delta_{\omega_i,\land}(0) \cdot sP$$

$$= sY$$

(10)

4. Comprehensive Performance Analysis

4.1 Security Analysis

It is trivially guaranteed that unauthorized data receivers have no access to the plaintext. Anyone that does not hold authorized attributes cannot extract $sY$. Accordingly, such a data receiver cannot recover $H(sY)$.

**Theorem.** If CDH assumption is hard, our scheme is secure under chosen ciphertext attacks (CCA-secure) in the random oracle model.

**Proof.** Due to page limitation, we do not present the proof of the theorem. Interested readers can find the completed proof in the full version of this paper.

4.2 Expressiveness Analysis

4.2.1 Equivalence. Our weighted mechanism is a generalized form of traditional threshold mechanism. So, we can transform the “AND/OR” gate to a generalized form based on the weighted mechanism. For example, there are some data receivers that can decrypt some ciphertexts associated with both attribute “A” and attribute “B”. For the threshold mechanism, the access policy of their private key is
defined as \((2, 2)\) to represent “A and B”. For the weighted mechanism, we define weights of “A” and “B” as, respectively, \(w_1\) and \(w_2\). In this case, the access policy of their private keys is defined as \((w_1 + w_2, w_1 + w_2)\) to represent “A and B”.

If there are some data receivers that can decrypt some ciphertexts associated with attribute “A” or attribute “B”. For the threshold mechanism, the access policy of their private key is defined as \((1, 2)\) to represent “A or B”. For the weighted mechanism, the access policy of their private keys is defined as \((\min\{w_1, w_2\}, w_1 + w_2)\) to represent “A or B”. Therefore, the proposed generalization is feasible. We present this expressiveness equivalence in Figure 2.

![Figure 2](image1.png)

**Figure 2** The expressiveness equivalence between the threshold mechanism and the weighted mechanism.

4.2.2 Simplicity. Compared to previous scheme, our weighted mechanism provides superior expressiveness with a simple structure. For example, there are some data receivers that can decrypt some ciphertexts associated with some attribute combinations such as “AB” or “AC” or “BCD”. If we use “AND/OR” gates to express this relation, the access policy can be defined as:

\[(A \text{ and } (B \text{ or } C)) \text{ or } (B \text{ and } C \text{ and } D)\]  \hspace{1cm} (11)

For the weighted mechanism, we can define weights of A, B, C, and D as, respectively, 4, 3, 2, 1. In this case, the access policy can be defined as \((6, 10)\), which is adequate to represent this relation. We show this simplicity of access policy in Figure. 3. Therefore, the proposed mechanism is very efficiency in terms of expressiveness.

![Figure 3](image2.png)

**Figure 3** The simplification of an access policy by the weighted mechanism.

4.3. Computational Efficiency Analysis

In this section, we discuss the computational efficiency of previous schemes [2], [3], [6], [13] and our KP-WABE-WB. Let \(\mathbb{G}\) be the bit size of an element in \(\mathbb{G}\), \(|\mathcal{A}|\) be the total number of attributes in global attribute space, \(|\mathcal{P}|\) be the total numbers of attributes in an access policy, \(|\mathcal{S}|\) be the total numbers of attributes in an attribute set, and \(z_q^i\) be the bit size of an element in \(z_q^i\). The detailed
comparison is listed in Table I in four aspects, which are the size of public parameter \( |\mathcal{S}_K| \), the size of private key \( |\mathcal{S}_K| \), the size of ciphertext \( |CT| \), and the total times of bilinear pairing computation \( N_{bp} \).

**Table 1** Computational Efficiency Comparison

| Schemes | \( |\mathcal{S}_K| \) | \( |CT| \) | \( N_{bp} \) |
|---------|----------------|----------------|-------------|
| [2]     | \( |\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( |CT| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( |\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) |
| [3]     | \( 2|\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( 2|\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( 2|\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) |
| [6]     | \( |\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( |\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( |\mathcal{S}_K| \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) |
| [13]    | \( \mathcal{S}_K \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( \mathcal{S}_K \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( \mathcal{S}_K \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) |
| Our scheme | \( \mathcal{S}_K \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | \( \mathcal{S}_K \) \( + \) \( \mathcal{G}_1 \) \( \times \) \( \mathcal{G}_2 \) | 0 |

As is summarized in Table I, the size of private key and ciphertext in our scheme is trivial among those schemes. For Qin et al.’s [13] construction, an attribute matrix with \( l \) rows and \( d \) columns is introduced to show hierarchy of different attributes. Meanwhile, their construction is based on composite-order bilinear pairing so the computational efficiency is a little bit unacceptable. Compared to those schemes in Table I, our KP-WABE-WB is the only one that need no bilinear pairing computation. Therefore, it is efficient to build secure cloud data sharing via our KP-WABE-WB.

5. Conclusion
Attribute-based encryption (ABE) is an interesting cryptographic technique for flexible cloud data sharing access control. In this paper, we propose a key-policy weighted attribute-based encryption without bilinear pairing computation (KP-WABE-WB) for secure cloud data sharing access control. A simple weighted mechanism is presented to describe different importance of each attribute, which not only enhances the expressiveness of access policy but also simplifies its structure. We introduce a novel construction of ABE without executing any bilinear pairing computation. Therefore, the computational efficiency is markedly improved. Compared to previous schemes, our scheme has a better performance in expressiveness of access policy and computational efficiency.

Our future work will build on the preliminary construction in this work to develop the proposed scheme. Considering some specific industrial scenario including secure data sharing in NB-IoT, we need to design a concrete algorithm of access policy generation based on weighted attribute.

6. Acknowledgment
This work is supported by the National Natural Science Foundation of China under Grant 61672299, Grant 61373135, and Grant 61170276. We thank the anonymous reviewers for their enlightening comments.

7. References
[1] A. Sahai and B. Waters, “Fuzzy Identity-Based Encryption,” in *Proc. EUROCRYPT*, 2005, pp. 457-473.
[2] V. Goyal, O. Pandy, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in *Proc. ACM CCS*, 2006, pp. 89-98.
[3] J. Bethencourt, A. Sahai, B. Waters, “Ciphertext-Policy Attribute-Based Encryption,” in *Proc. IEEE Symp. Secur. Privacy*, 2007, pp. 321-334.
[4] L. Cheung and C. Newport, “Provably secure ciphertext policy ABE,” in *Proc. ACM CCS*, 2007, pp. 456-465.
[5] B. Waters, “Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization,” in *Proc. Public Key Cryptogr.*, 2011, pp. 53-70.
[6] M. Green, S. Hohenberger, and B. Waters, “Outsourcing the Decryption of ABE Ciphertexts,” in *Proc. USENIX Secur. Symp.*, 2011, pp. 34.
[7] J. Lai, R. H. Deng, C. Guan, and J. Weng, “Attribute-Based Encryption With Verifiable Outsourced Decryption,” *IEEE Trans. Forensics Security*, vol. 8, no. 8, pp. 1343-1354, 2013.

[8] S. Q. Lin, R. Zhang, H. Ma, and M. S. Wang, “Revisiting Attribute-Based Encryption With Verifiable Outsourced Decryption,” *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 10, pp. 2119-2130, 2015.

[9] D. Pletea, S. Sedghi, M. Veeningen, and M. Petkovic, “Secure distributed key generation in attribute based encryption systems,” in *Proc. ICITST*, 2015, pp. 103-107.

[10] X. L. Xu, J. L. Zhou, X. H. Wang, and Y. Zhang, “Multi-authority proxy re-encryption based on CPABE for cloud storage systems,” *Journal of Systems Engineering and Electronics*, vol. 27, no. 1, pp. 211-223, 2016.

[11] J. Hur, “Improving Security and Efficiency in Attribute-Based Data Sharing,” *IEEE Trans. Knowl. Data. Eng.*, vol. 25, no. 10, pp. 2271-2282, 2013.

[12] N. Oualha and K. T. Nguyen, “Lightweight Attribute-Based Encryption for the Internet of Things,” in *Proc. ICCCN*, 2016, pp. 1-6.

[13] B. Qin, H. Deng, Q. H. Wu, J. Domingo-Ferrer, D. Naccache, and Y. Y. Zhou, “Flexible attribute-based encryption applicable to secure e-healthcare records,” *International Journal of Information Security*, vol. 14, no. 6, pp. 499-511, 2015.