JES
Journal of Engineering Sciences

Vol 13, Issue 01, Jan /2022
ISSN NO:0377-9254

IMPROVING SECURITY AND PRIVACY ATTRIBUTE BASED DATA SHARING IN CLOUD COMPUTING

[1]Kadali Madhu Priya, [2]Mrs.P Chaitanya

[1]M.Tech [CSE], Eluru college of Engineering and Technology, Duggirala - 534004, Andhra Pradesh.

[2]Professor, Eluru college of Engineering and Technology, Duggirala - 534004, Andhra Pradesh.

Abstract—Data sharing is a convenient and economic service supplied by cloud computing. Data contents privacy also emerges from it since the data is outsourced to some cloud servers. To protect the valuable and sensitive information, various techniques are used to enhance access control on the shared data. In these techniques, Ciphertext-policy attribute-based encryption (CP-ABE) can make it more convenient and secure. Traditional CP-ABE focuses on data confidentiality merely, while the user's personal privacy protection is an important issue at present. CP-ABE with hidden access policy ensures data confidentiality and guarantees that user's privacy is not revealed as well. However, most of the existing schemes are in efficient in communication overhead and computation cost. Moreover, most of those works take no consideration about authority verification or the problem of privacy leakage in authority verification phase. To tackle the problems mentioned above, a privacy preserving CP-ABE scheme with efficient authority verification is introduced in this paper. Additionally, the secret keys of it achieve constant size. Meanwhile, the proposed scheme achieves the selective security under the decisional n-BDHE problem and decisional linear assumption. The computational results confirm the merits of the presented scheme.

Index Terms—Attribute-based encryption (ABE), authority verification, hidden access policy, privacy preserving.

I. INTRODUCTION

CLOUD techniques make it possible to utilize information technology resources into business domain. The cloud provides variety of scalable services on-demand, such as online databases, program interface, storage and computing resources, etc. Users can obtain services through phones, laptops, and desktops as shown in Fig. 1. Cloud storage provides remote data storage and management services. It is also helpful in data analyzing and computing, which is quite simple as it can provide a variety of services at the same time. Cloud has many advantages in data storage, such as decreasing communication cost and maintenance charge, saving resources, allowing remote access, and so on. However, people might not be willing to store

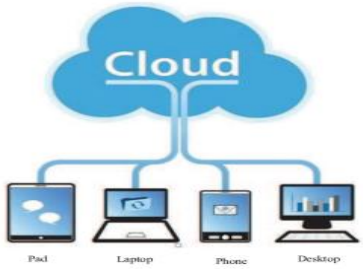

Fig. 1. Cloud computing model.

their data in the cloud, even though it provides so many benefits because of the data confidentiality and privacy problems. The cloud server (CS) may be untrusted, in other words, if data is uploaded to cloud, the cloud service provider may obtain and disclose users' personal privacy, and even access and share the data illegally [1].

To make sure the confidentiality of the data in cloud, people are inclined to encrypt them before they are uploaded to cloud. But the general encryption algorithms make the data process become difficult. ABE is a good candidate to overcome this limitation. ABE was first proposed in 2005 by Sahai and Waters [2], which guaranteed the data confidentiality and provided the fine-grained access control policy to the customers. It has been widely accepted as an effective method encrypting the outsourced data in cloud computing. ABE improves the efficiency when the data owner (DO) intends to share data contents with multiusers. It permits DO to specify an access policy to the encrypted files, which can make the users who match it, access uploaded data. The users who do not satisfy the access structure cannot get any information about the data contents. For instance, we consider the data access control for a company. If the CEO intends to submit a classified file, through the cloud, to the managers in sales department, planning department, and research and development (R&D) department. Then he/she can use an ABE scheme. First he/she encrypts the file and specifies an access structure as ω = manager ∧ (sales department ∨ planning department ∨ R&D). Next he/she uploads the encrypted file and the access structure into the CS. Only the managers in the three mentioned departments can access the classified file, and the managers in other departments or the general staff in the three mentioned departments cannot learn anything about the file even if they collude.

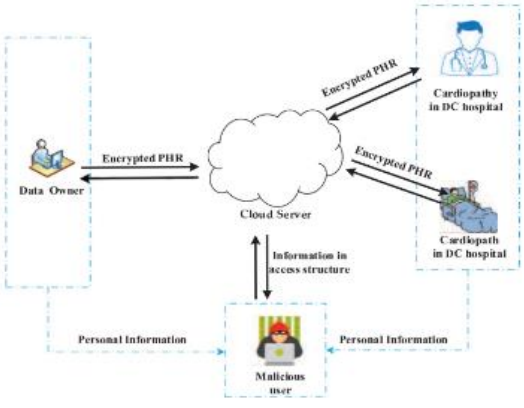

Fig. 2. Privacy leakage model.

Most of ABE proposals perform very well in secure data sharing. However, the personal privacy of the DO and the users is ignored in these constructions. For convenience of recovering data, the access policy is always sent with ciphertexts. In some scenarios, the access structure may carry sensitive information of users. For instance, a patient wants to share his/her personal health record (PHR) with some doctors and family members, but he/she may not want others to know that he/she is sick. If the patient employs a normal ABE scheme to encrypt the PHR, although the malicious user cannot get the contents of the PHR, he/she may get some information about the users as shown in Fig. 2. The access policy contains "cardiopathy" and "DC hospital" and the malicious third party may guess that the DO is suffering from a heart attack and is treating in the DC hospital. Hence a natural problem is how to keep the shared data secure, while the privacy of them is also protected.

II. LITERATURE SURVEY

ABE schemes specify two kinds of policies, the key policy (KP) and the ciphertext policy (CP), which result in a KP-ABE and a CP-ABE, respectively. In a KP-ABE scheme, the private key is generated under a specified access policy, while in a CP-ABE scheme, the ciphertext is related to a specified access policy. Only users whose attributes match the policy can re- cover data successfully. Now ABE has been a hot research area [3]–[7]. However, most of them take only the data confidentiality into

consideration while ignoring the importance of user privacy preserving.

The first work with consideration of user personal privacy was introduced by Nishide et al. [8], where the access policy was partially hidden by dividing attribute into two parts as value and name, while only hiding the value. Due to the hidden policy, the adversary cannot get any information about the users. However, their scheme is impractical since its computation cost is too high. In 2009, Waters proposed a CP-ABE scheme with dual system encryption technique [7]. It provided a new way for privacy preserving in CP-ABE. Then Lai et al. [9], [10] used this technique to issue two hidden access policy CP-ABE schemes (HP-CP-ABE). Both of them have been proven to achieve full security. The first one [9] only supports AND gate, and the second one [10] supports linear secret share scheme (LSSS) [11], which is a more expressive access structure. However, the size of both secret keys and ciphertext increases linearly with the number of attributes. Then Rao et al. [12] introduced another HP-CP-ABE scheme with full security. In this scheme, its security also relies on composite-order group, but the size of secret keys and ciphertext achieves constant which improves the efficiency compared with [9] and [10]. However, this scheme only supports AND gate, which is not expressive. Zhang et al. [13] proposed a hierarchical HP-CP-ABE scheme, where they used the technique proposed by Abdalla et al. [14]. It achieves constant size secret keys and supplies fast decryption. Recently, Huang et al. [15] presented an HP-CP-ABE with lower computation cost and constant size secret keys. However, it only achieves selective security, which is not a strong enough security model. Although the above-mentioned schemes can protect users' privacy, there is an important problem to be ignored. That is to say, if the access policy is hidden, the users have to attempt the entire possible combinations of the secret keys to decrypt the ciphertexts, which means the users must take more time to recover messages. It is necessary to find a method to help the users decrypt ciphertexts efficiently and successfully. To address this problem, Zhang et al. [16] introduced an HP-CP-ABE scheme with authority verification phase to decrease users' computational consumption. The authority verification phase can help users check whether they are the valid users or not. However, privacy leakage is found in the match phase. Then Li et al. [17] proposed a more efficient HP-CP-ABE scheme with authority verification. It can decrease users' unnecessary computational consumption. However, the same problem with [16], attributes in access policy can also be tested out in authority verification phase. Cui et al. [18] introduced another HP-CP-ABE scheme. But the size of secret keys and ciphertexts are both increasing with the number of attributes. Recently, Khan et al. [19] proposed an HP-CP-ABE scheme with LSSS access structure. And it also supports authority verification. However, they employed hidden vector encryption to achieve this. It is not efficient enough, relatively. Zhang et al. [20] presented an HP-CP-ABE scheme, where it supports large universe attribute set and LSSS access policy.

However it is based on composite order group, which is inefficient than the schemes based on prime order group in the same case. Another technique supporting hidden access policy comes from inner-product predicate encryption (IPE) [21], [22]. However, this conversion will generate a great loss in efficiency. An instance based on IPE due to Phuong et al. [23] shows that the size of secret keys and ciphertexts in this scheme increases linearly with the depth of attributes, which takes more storage and computational resource.

III. methodology

3.1 Motivation

Efficient decryption test is indispensable and necessary, which is used to help users decrypt quickly and successfully, and make hidden access policies, CP-ABE schemes to be more practical. Additionally, in some schemes, the size of secret keys or cipher-texts is too long. It is not friendly with storage for light devices as nowadays mobile devices are becoming more and more popular. This increase computation cost of users and it is also a waste of resource as well.In a word, to keep security and privacy preserving of the shared data in cloud, the following emergency issues must be addressed, simultaneously:

1) data confidentiality;

2) privacy preserving;

3) efficient decryption test;

4) efficiency, such as parameters size and time consumption of algorithms.

From the analyses of related works, one can find that all existing schemes cannot provide a comprehensive solution to the above-mentioned problems. These motivate us to construct such an HP-CP-ABE scheme that can achieve privacy protection, support efficient authority verification, and have lower computation cost.

3.2 System Model and Framework

1) System Model: There are four entities in a HP-CP-ABEsystem: A CS, an authority center (AC), DO, and data users (DU) as shown in Fig. 3.

1) AC: In the HP-CP-ABE, it should be fully trusted and accepts the registration of all DU. Then it will generate public keys and secret keys for each DU.

2) DO: DO specifies access policies and encrypts data. Then he/she uploads the encrypted data to the CS.

3) CS: CS may not be honest in the system. It is in charge of storing encrypted data.

4) DU: DU can request secret keys associated with their attributes from AC and access to encrypted data from CS. If DU can pass the verification, which means their attributes match the policy, then DU can recover the encrypted contents.

Note that in this situation, the access policies are hidden when DO encrypt data, thus no one can get any information from the ciphertext.

2) Framework: A HP-CP-ABE scheme is given as follows. 1) Setup(κ, U) → (PK, MK): Let κ, U denote the security parameter and universe of attribute. This algorithm takes as input κ,U and outputs the public keys PK and the master key MK.

2) KeyGen(PK, MK, L) → SKL: Given PK MK, and a subset L ⊂ U, this algorithm generates the corresponding private key SKL. 3) Encrypt(PK,M,W) → CT: Given PK, data M, and the

policy W, this algorithm outputs the encrypted content

CT.

4) Decrypt(PK, SK,CT) → M or ⊥: Given PK, CT, if DU's attribute satisfies W, this algorithm outputs the plain text M.

3.3 . Security Model

The proposed scheme is secure under the selective policies. In this security model, an

adversary A interacts with a simulator B, and they run the game as follows.

1) Init: A announces W* 0 and W* 1 as two challenge access policies.

2) Setup: B runs Setup algorithm and delivers PK to A.

3) Phase 1: A queries secret key for an attribute list L, if L |= W* 0 ∧ L |= W*1 or L W* 0 ∧ L W*

1 . Then B re-turns secret key SKL to A. And A can repeat the queries for polynomial times.

4) Challenge: A submits two equal length challenge plain- text M0 and M1. If L |= W* 0 ∧ L |= W* 1 in Phase1 then M0 = M1. Otherwise B chooses b ∈ {0, 1} and returns Encrypt(PK, Mb, W* b) to A.

5) Phase 2: The operations in Phase 1 is repeated.

6) Guess: A outputs a guess b ∈ {0, 1} of b.

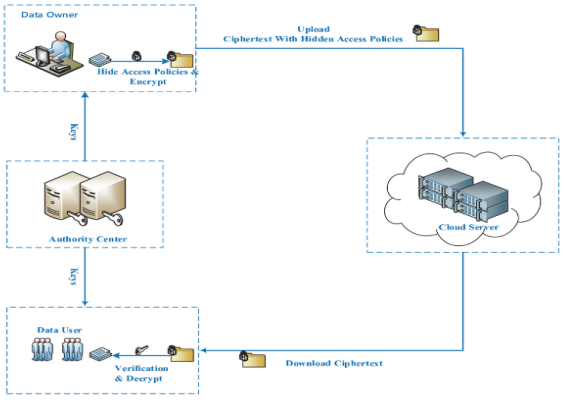

Fig 3System model.

IV. Algorithms used

Algorithm 1: IdentifierGen Algorithm.

Input:

1: DU's attribute list $L = \{L_1, L_2, \ldots, L_n\}$.

2: Convert each attribute into a binary array as $L_i = l_{i,1} l_{i,2}, \ldots, l_{i,n}$ for all $L_i \in L$. Then L can be described as $L = l_{1,1} l_{1,2} \ldots l_{1,n} \ldots l_{n,1} l_{n,2} \ldots l_{n,n} (l_{i,j} \in \{0,1\})$. For convenience, let $L = l_1 l_2 \ldots l_k (k = n^2, l_i \in \{0,1\})$.

3: AC picks $\alpha_i, \beta_i \in_R \mathbb{Z}_p$ and generates DU's identifier as follows. Set $w_0 = g$.

4: For $i = 1$ to k, AC computes

$$w_i = w_{i-1}^{\alpha_i^{l_i} \beta_i^{1-l_i}}.$$

Output:

5: $w_k = w_{k-1}^{\alpha_k^{l_k} \beta_k^{1-l_k}}$.

Algorithm 2: KeyGen Algorithm.

Input:

1: DU's attribute list $L = \{L_1, L_2, \ldots, L_n\}$.

2: Run **IdentifierGen** algorithm.

3: Define a function $H_n(L) = h_0 \prod_{i=1}^{n} T_{i,j} = h_0 \prod_{i=1}^{n} h_i^{a_{i,j}}$.

4: Pick $r_1, r_2 \in_R \mathbb{Z}_p$.

Output:

5: Secret keys for L are given as follows.

$$D_0 = K_L = w_k, \quad D_1 = u H_n(L)^{r_1} \omega^{r_2}$$
$$D_2 = g^{r_1 t_1 t_2 + r_2 t_1 t_3}, \quad D_3 = g^{r_1 t_1}$$
$$D_4 = g^{r_2 t_1}.$$

Note: all $T_{i,j}$ are public keys.

Algorithm 3: Authority Identification.

1: DU makes a request to CS.

2: CS delivers C_0 to the DU.

3: DU computes $e(D_0, C_0)$ and returns it to CS.

4: CS checks:

If $e(D_0, C_0) = C_k'$, where $i \in [1, t]$, CS returns C_k'' and CT_1 to DU. Then DU runs Decryption algorithm.

Else CS rejects DU's request and aborts the algorithm.

Algorithm 4: Decryption.

1: **Compute**: $m_k = e(D_0, C_0) \oplus C_k''$

Input: $m_k, CT_1, D_1, D_2, D_3, D_4$.

2: **Compute**:

$$M = C_5 \Big/ \frac{e(D_1, C_1) \cdot e(D_2, C_2)}{e(D_3, \prod_{i=1,(i,j)\in m_k}^{n} C_{i,j}) \cdot e(D_4, C_4)}.$$

Output: The message M.

V. PERFORMANCE COMPARISONS AND EXPERIMENTS ANALYSIS

A. Performance Comparisons

In this section, we compare our work with previous works [8], [10], [12], [16], [17], and [20] in the aspect of security feature and performance. In Table I, we give comprehensive comparisons about some indispensable security features, such as privacy preserving, decryption test, group order, access policy type, and security model. From Table I, we can know that all the schemes realize policy hiding, but only schemes in [16], [17], and [20], and our scheme support decryption test. Unfortunately, privacy leakage is found in decryption test phase of [16] and [17]. Scheme in [20] supports LSSS access structure and is quite expressive. However its security relies on composite group. Details comparisons among [16], [17], and [20], and our scheme will be given in Table II.

In Table II, PK, MK, SK, and CT denote public parameter, master key, secret key, and ciphertext, respectively. Let U = {Att1, Att2,..., Attn} be the universe of the attributes. ki is the number of values in Atti. K = n i=1 ki is the total number of all the values in U. N is the number of attributes in access policy. We use Ce and CT to denote a bilinear operation and an

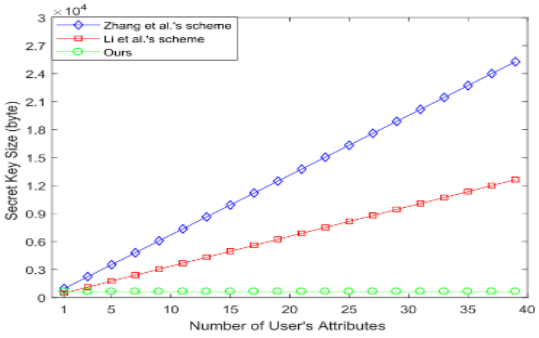

Fig. 4. Secret key size.

exponentiation operation in GT . |G|, |GT |, and |Zp| denote the bit length of the elements belongs to G, GT , and Zp, where p is the group order. I denotes the minimum authorized user set and its size is |I|. From Table II, we can know that only our scheme realizes constant size secret key and has the least pair operation in decryption test phase. Of course, the PK size in our scheme is larger, but the CT size and decryption cost are smaller than other schemes.

VI. CONCLUSION

We proposed a privacy preserving CP-ABE scheme in the standard model. The presented scheme has many advantages over the existing schemes, such as constant size private keys and short ciphertexts. And in decryption, it only needs four pairing computations. The proposed scheme achieves selective security and anonymity in a prime order group. In the standard model, we show the security of the proposed scheme is reduced to the decisional n-BDHE and the DL assumptions. Additionally, the proposed scheme supports authority verification with no privacy leakage. However, the introduced scheme only supports “AND” policy and relies on a weak security model. How to construct a strong secure HP-CP-ABE scheme with more flexible access policy is left for the future works.

REFERENCES

[1] P. P. Kumar, P. S. Kumar, and P. J. A. Alphonse, "Attribute based encryptionin cloud computing: A survey, gap analysis, and future directions," J. Netw. Comput. Appl., vol. 108, pp. 37–52, 2018.

[2] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Proc. 24th Annu. Int. Conf. Theory Applications Cryptographic Techn., May 2005, vol. LNCS 3494, 2015, pp. 457–473.

[3] K. Emura, A. Miyaji, A. Nomura, K. Omote, and M. Soshi, "A ciphertext- policy attribute-based encryption scheme with constant ciphertext length,"in Proc. 5th Int. Conf. Inf. Security Practice Experience, Apr. 2009, pp. 13–23.

[4] J. Han, W. Susilo, Y. Mu, and J. Yan, "Privacy-preserving decentralized key-policy attribute-based Encryption," IEEE Trans. Parallel Distrib.Syst., vol. 23, no. 11, pp. 2150–2162, Nov. 2012.

[5] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," IEEE Trans. Inf. Forensics Secur., vol. 11, no. 6, pp. 1256–1277, Jun. 2016.

[6] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in Proc. 30th Annu. Int. Conf. Theory Appl. Cryptographic Techn.: Advances Cryptology, May 2011, pp. 568–588.

[7] B. Waters, "Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions," in Proc. 29th Annu. Int. Cryptology Conf. Advances Cryptology, Aug. 2009, pp. 619–636.

[8] T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in Proc. Appl. Cryptogr. Netw. Security, Jun. 2008, vol. LNCS 5037, pp. 111–129.

[9] J. Lai, X. Zhou, R. H. Deng, and Y. Li, "Fully secure cipertext-policy hiding CP-ABE," in Proc. 6th ACM Symp. Inf. Comput. Commun. Secur., 2011, pp. 24–39.

[10] J. Lai, X. Zhou, R. H. Deng, Y. Li, and K. Chen, "Expressive CP-ABE with partially hidden access structures," in Proc. 7th ACM Symp. Inf. Comput. Commun. Secur., May 2012, pp. 18–19.

[11] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in Proc. 14th Int. Conf. Prac- tice Theory Public Key Cryptography Conf. Public Key Cryptography, Mar. 2011, pp 53–70.

[12] Y. S. Rao and R. Dutta, "Recipient anonymous ciphertext-policy attribute based encryption," in Proc. 9th Int. Conf. Inf. Sys. Secur., Dec. 2013, pp. 329–344.

[13] L. Zhang, Q. Wu, Y. Mu, and J. Zhang, "Privacy-preserving and secure sharing of PHR in the cloud," J. Med. Syst., vol. 40, pp. 1–13, 2016.

[14] M. Abdalla, D. Catalano, and D. Fiore,"Verifiable random functions: Relations to identity-based key encapsulation and new constructions," J. Cryp- tol., vol. 27, pp. 544–593, 2014.

[15] C. Huang, K. Yan, S. Wei, G. Zhang, and D. H. Lee, "Efficient anonymous attribute-based encryption with access policy hidden for cloud computing," in Proc. IEEE Int. Conf. Progress Inform. Comput., Dec. 2017, pp. 266– 270.

[16] Y. Zhang, X. Chen, J. Li, D. Wong, and H. Li "Anonymous attribute-based encryption supporting efficient decryption test," in Proc. 8th

ACM Symp. Inf. Comput. Commun. Secur., May 2013, pp. 511–516.

[17] J. Li, H. Wang, Y. Zhang, and J. Shen, "Ciphertext-policy attribute-based encryption with hidden access policy and testing," KSII Trans. Internet Inf. Syst., vol. 10, no. 7, pp. 3339–3352, Jul. 2016.

[18] H. Cui, R. H. Deng, G. Wu, and J. Lai, "An efficient and expressive Ciphertext-policy attribute-based encryption scheme with partially hidden access structures," in Proc. 10th Int. Conf. Prov. Secur., Nov. 2016, pp. 19–38.

[19] F. Khan, H. Li, L. Zhang, and J. Shen, "An expressive hidden access policy CP-ABE," in Proc. IEEE 2nd Int. Conf. Data Sci. Cyberspace, Jun. 2017, pp. 26–29.

[20] Y. Zhang, Z. Dong, and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," IEEE Int. Things J., vol. 5, no. 3, pp. 2130–2145, Jun. 2018.

[21] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in Proc. 29th Annu. Int. Conf. Theory Appl. Cryptographic Techn., 2010, pp. 62–91.

[22] T. Okamoto and K. Takashima, "Adaptively attribute-hiding (hierarchical) inner product encryption," in Proc. 31st Annu. Int. Conf. Theory Appl. Cryptographic Techn., May 2012, pp. 591–608.

[23] T. V. X. Phuong, G. Yang, and W. Susilo, "Hidden ciphertext policy attribute-based encryption under standard assumptions," IEEE Trans. Inf. Forensics Secur., vol. 11, no. 1, pp. 35–45, Jan. 2015.

[24] X. Boyen and B. Waters, "Anonymous hierarchical identity-based encryption (without random oracles)," in Proc. 26th Annu. Int. Conf. Advances Cryptology, Aug. 2006, pp. 290–307.

[25] J. H. Park and H. L. Dong, "Anonymous HIBE: Compact construction over prime-order groups, "IEEE Trans. Inf. Theory, vol. 59, no. 4, pp. 2531– 2541, Apr. 2013.

[26] J. H. Seo, T. Kobayashi, M. Oukubo, and K. Suzuki, "Anonymous hierarchical identity-based encryption with constant size ciphertexts," in Proc. Int. Conf. Practice Theory Public Key Cryptography, Mar. 2009, vol. 5443,pp. 215–234.

[27] F. Li and W. Wu, Pairing-Based Cryptography. Beijing, China: Science Press, 2014.