Multiowner Data Sharing using Block Chain

Dr. A. Kanchana[1], S. Aasha[2], K. Priyadharshini[3], B. Narmatha[4], S. Keerthana[5]

[1] Professor, Department of CSE, Mahendra Engineering College for Women, Kumaramangalam.
Email: hodcse@mecw.org

[2, 3, 4, 5] UG Scholars, Department of CSE, Mahendra Engineering College for Women,
Kumaramangalam. Email: [4]narmibhuvanesh07@gmail.com

Abstract - Ciphertext-Policy Attribute-Based Keyword Search (CP-ABKS) facilitates search queries and supports fine-grained access control over encrypted data in the cloud. However, prior CP-ABKS schemes were designed to support *unshared* multi-owner setting, and cannot be directly applied in the *shared* multi-owner setting (where each record is accredited by a fixed number of data owners), without incurring high computational and storage costs. In addition, due to privacy concerns on access policies, most existing schemes are vulnerable to off-line keyword-guessing attacks if the keyword space is of polynomial size. Furthermore, it is difficult to identify malicious users who leak the secret keys when more than one data user has the same subset of attributes. In this paper, we present a privacy-preserving CP-ABKS system with hidden access policy in *Shared* Multi-owner setting (basic ABKS-SM system), and demonstrate how it is improved to support malicious user tracing (modified ABKS-SM system). We then prove that the proposed ABKS-SM systems achieve selective security and resist off-line keyword-guessing attack in the generic bilinear group model. We also evaluate their performance using real-world datasets.

I. INTRODUCTION

CLOUD computing is widely used by both individuals and organizations (including government agencies), for example to store and process large volume of data (e.g., text, image, and video), which are typically encrypted prior to outsourcing. Searchable Encryption (SE) schemes enable data users to securely search and selectively retrieve records of interest over encrypted data (outsourced to the cloud), according to user-specified keywords. There are, however, other desirable properties when dealing with encrypted data outsourced to the cloud. For example, when encrypting significant volume of data, conventional encryption approaches suffer from limitations due to having multiple copies of ciphertexts (e.g., in public key encryption schemes) and complex and expensive key management (e.g., in symmetric encryption schemes). Ciphertext-Policy Attribute-Based Encryption (CP-ABE) schemes are designed to mitigate these two limitations, as well as enhancing access permissions in multi-user setting and facilitating one-to-many encryption. However, in standard CP-ABE schemes, an access policy in plaintext is associated with a ciphertext may result in leakage of sensitive information. For example, in an e-health system, hospital A encrypts a patient's electronic medical record (EMR) using CP-ABE with an access policy, such as ("ID: 1788" AND "Hospital: Hospital A") OR ("Doctor: Cardiologist" AND "Hospital: Hospital B") Hence, one can easily infer from the user attribute set ("Cardiologist"," Hospital B") that patient ("ID: 1788") in Hospital A likely suffers from a heart condition. Such privacy leakage is clearly not appropriate, particularly if the medical condition is more sensitive (e.g., sexually transmitted diseases such as chlamydia, gonorrhea, and human papillomavirus infections). In addition, medical organizations are subject to exacting regulatory oversight in most developed jurisdictions. Hence, there have been efforts to design CP-ABE scheme with hidden access policies.

However, in many applications, data records are co-owned by a number of data owners, rather than a single data owner. That is to say, each file is encrypted by multiple data owners, and the data user can access the file, if and only if, he/she obtains authorizations from several data owners. For example, the EMR for a certain patient is controlled by multiple departments (e.g., clinical departments such as infectious diseases and psychiatry) and/or medical organizations (e.g., San Antonio Behavioral Healthcare Hospital, Texas Center for Infectious Disease, and Texas Infectious Disease Institute). Deploying CP-ABKS schemes , in the unshared multi-owner setting (where multiple data owners manage different data records) incur significant computational and storage costs. Another realistic, but more complex, setting is the shared multi-owner setting, where each record is co-owned by multiple data owners. The differences between unshared multiowner setting and shared multi-owner setting are described.

II. RELATED WORK

The first symmetric SE scheme and asymmetrical SE scheme were presented by Song et aland Boneh et al., respectively. Subsequent SE schemes were designed to support a range of features, such as single keyword search, multi-keyword search and ranked keyword search , CP-ABE was designed to allow fine-grained access control over ciphertexts, and CP-ABKS was designed to support both fine-grained access control and keyword search simultaneously. For example, presented the CP-ABKS scheme that enables data owners to grant fine-grained search permissions, Sun et al. presented an owner-enforced CP-ABKS scheme that supports user revocation and is shown to be selectively secure against chosen-keyword attack. However, the computational costs of these two schemes grow linearly as the number of system attributes increases. This is not scalable in practice. To minimize computational costs and ciphertext size required in such schemes,

implemented a keyword search function in attribute-based encryption (ABE) scheme, by outsourcing key-issuing and decryption operations. Dong et al also designed an efficient CP-ABKS scheme via an online/offline approach when considering resource constrained mobile devices. One serious limitation of CP-ABE schemes is that the access policy embedded in the ciphertexts may leak sensitive information to authorized data users, as discussed in the preceding section. Thus, constructed a more practical CP-ABE scheme, which allows the encryptor to use wildcards to represent certain attributes in a hidden solution. Similarly ,a hidden access policy scheme, which supports AND-gate with wildcard by utilizing inner product encryption. These prior CPABE schemes with partially hidden access policy have high computational costs and do not support keyword search over encrypted data. To resist off-line keyword-guessing attacks, a secure CP-ABKS scheme supporting keyword search and hidden access structure. Also, as discussed earlier, such schemes generally consider only *unshared* multi-owner setting. For example, Zhang et al. provided privacy-preserving ranked multi-keyword search in the multi-owner model and prevented attackers from eavesdropping secret keys .designed an efficient multi-keyword search scheme with fine-grained access control. Should these schemes be deployed in a *shared* multi-owner setting, they will need the same random parameter for each individual data owner, which clearly is impractical in practice particularly as the number of data owners increases.

III.BLOCKCHAIN TECHNOLOGY

The basic concept of block chain was proposed by Nakamoto Satoshi . Blockchain is a decentralized database using cryptographic technology to generate associated blocks, where each block records full transactions over a period of time. Each node contains a complete historical block, and even if one node is modified, it will not affect the verification of entire block chain. The block chain information is public, and anyone can search the chain for historical trading information. Block chains require miners, when any node in block chain generates a transaction, the transaction is broadcast to each miner, and all miners may verify the transaction through the proof of work mechanism. Blockchain has the following main features: (i) the ability of anti-modification (ii) the ability of toleranceas some nodes are faulty (iii) the ability of reaching collaborative trust among nodes in this distributed peer-to peer system without the third-party certification agency (iv) the ability of accessing information of block chain at any node in this network. A block chain consists of continuous blocks, each block records and stores serial transections as a Merkle root by using Merkle tree algorithm in a period of time, and every block after the first block (called genesis block) has acryptographic hash value of previous block (called parent block). All these hashed values are assembled into a chain, On the other hand, the private blockchain has no mining mechanism and miner's role, because the private chain is usually used as a network within company,organization and members. For example, IBM block chain platform provides a private chain for more than 400 cooperatives, Wal-Mart offers solutions that provide traceability of food tracking through block chain to make food supply chain safety, and Maersk builds a global platform to achieve efficient transportation. In fact, the block chain is widely used in various intended applications , e.g., in Internet of Things or use for small mobile payments cloud computing digital certificate and other application scenarios.

Shared multi-owner setting. Both ABKS-SM systems consider the shared multi-owner setting and enable data owners to provide enhanced access control over their shared data with multiple permissions.

Hidden access policy. Both ABKS-SM systems provide hidden access policy, so that the access structure attached to the ciphertexts does not leak sensitive information about the encrypted data and its privileged recipients.

Tracing of malicious data users. To prevent dishonest data users from leaking their secret keys to others (e.g., for profits), the modified ABKS-SM system provides traceability by securely embedding their identity information in the secret keys.

IV. PRELIMINARIES

Let G,GT be two multiplicative cyclic groups of prime order p, g denotes a generator of group G, and e be the bilinear map G × G → GT with several properties: (1) Bilinearlity. $e(\sim\ell 1\ 1\ ,\ \sim\ell 2\ 2\ ) = e(\sim 1, \sim 2)\ell 1\ell 2$ for all $\sim 1, \sim 2 \in$ G, $\ell 1, \ell 2 \in_R$ Zp; 2)Non-degeneracy. There are elements $\sim 1, \sim 2 \in$ G satisfying $e(\sim 1, \sim 2) \neq 1$; (3) Computability. There is an efficient algorithm to compute $e(\sim 1, \sim 2)$ for $\forall \sim 1, \sim 2 \in$ G. $x \in_R X$ is defined as choosing an element x uniformly at random from the set X, and denotes an integer set $\{1, 2, ...,\_\}$, where _ is an integer

V. LINEAR SECRET SHARING SCHEMES (LSSS)

Linear Secret Sharing Schemes (LSSS) converts any monotonic boolean formula into the LSSS representation, as well as enhancing access control based on multiple parties' requirements. The secret-sharing scheme _ over a group of parties $P = \{P1, P2, \cdots, Pl\}$ is called linear over field Zp if the following conditions hold.

• The shares for each party $Pi (i \in [1, l])$ form a vector over Zp. These elements $\{\omega i\}$ can be found in polynomial time in the size of the matrix$M$.

VI. GENERIC BILINEAR GROUP MODEL

We use oracles to execute the respective actions on G,GT and compute a non-degenerate bilinear map e : G × G → GT . We also use a random oracle to represent the hash function. In here, G is considered a generic bilinear group.

VII. SECURITY MODEL

In this section, we describe the security model for the basic ABKS-SM, based on the following security game .We also claim that the basic ABKS-SM system achieves selective security in the generic bilinear group model if there is no
2. For example, given an file *f* which includes the keyword *w*, 5 **Do**s first specify the file encryption key *kf* used to encrypt *f* as *c*, LSSS (*M*5×3) used to encrypt *kf* as *C*, access policy P used to encrypt *w* as *Iw*. If DU's attributes *Att* satisfy P, then the **CSP** can check whether the trapdoor *Tw′* matches with *Iw*. If these two conditions hold (*Att* |= P,*w* = *w′*), the **DU** gets the search results (*c*,*C*). However, the **DU** can
obtain *kf* , if and only if, he gains at least 3 decryption authorizations from 5 **DO**s.
probabilistic polynomial-time adversary *A* that can break the game with a non-negligible advantage. Note that the modified ABKS-SM system also achieves the selective security in the generic bilinear group model. Due to the space limitation, we omit the selective security of the modified ABKS-SM system. One would also note that the selective security goals mainly focus on the indistinguishability of
access policies and keywords. The selective security game for the basic ABKS-SM system is as follows:

• Setup: *A* selects two challenging access policies P0, P1 before sending them to *C*. After that, *C* first calls the **Setup** algorithm to generate the public key
PK and master key *MSK*, then it sends *PK* to *A* and keeps *MSK* itself.

• Phase 1: *A* picks an attribute list *Att* and issues the following oracle queries:
– *OKeyGenDU* (*Att*): If *Att* simultaneously satisfies both chosen access policies P0, P1, *C* runs **KeyGen***DU* to output the secret key *SK* before returning it to *A*. – *OTrap*(*Att*,*w′*): Given the submitted keyword *w′*, *C* executes **Trap** algorithm to generate the trapdoor (or search token) *Tw′* by leveraging *SK*, and then sends it to *A*.

• Challenge: *A* chooses two keywords *w*0,*w*1 ∈ *W* before returning them to *C*. If *A* gets access to *Tw′* on the condition that *Att* satisfies both access policies P0, P1 in Phase 1, we define *w*0 = *w*1. Then, *C* selects a random element *y* ∈ *{*0, 1*}* and uses **Enc** algorithm to generate the ciphertext *{Iwy }* by utilizing the corresponding P*y*. Finally, *C* sends *{Iwy }* to *A*.

• Phase 2: *A* repeatedly performs the operations in Phase 1. If *w*0 ≠ *w*1, then *A* cannot find *Att* that simultaneously satisfies P0, P1.

• Guess: *A* returns a guess bit *y′* ∈ *{*0, 1*}*, and *A* wins the security game if *y′* = *y*. *A*'s advantage ϵ in this selective security game is taken over the random bits used between *A* and *C*. Because *A* should conduct the challenging access policies P0, P1 before the Setup phase. This model is similar to the selective- ID model used in Identity-Based Encryption (IBE) schemes. However, the non-selective-ID model shown in CP-ABE scheme is proven secure in the generic bilinear group model. In the non-selective-ID security game, *A* can submit an attribute set *Att*, which satisfies both access policies P0, P1, and then *A* can obtain the corresponding search results. We further remark that *A* cannot gain sensitive information about P0, P1, except for the returned search results. This echoes the existing design of CP-ABE schemes with hidden access policy scheme .Generally, off-line keyword-guessing attacks are easier to conduct when keywords have low entropy. For example, keywords are chosen from a small keyword space, which allows an attacker to guess some candidate keywords in an off-line manner by utilizing the low-entropy characteristic of keywords. That is, given a trapdoor, an attacker can learn which keyword is used to generate the trapdoor as data user usually queries the commonly-used keywords with low entropy. Thus, to resist off-line keyword guessing attack, the above security definition also requires that malicious attackers should not be able to distinguish between the ciphertexts (or indexes) of two challenging keywords *w*0 and *w*1 of his/her choice.

Definition 1.

The basic ABKS-SM system achieves selective security in the generic bilinear group model, if there is an adversary *A* that can win above non-selective-ID security game with negligible advantage $\epsilon = |Pr[y' - y]$–12. Next, we present the traceability definition in the modified ABKS-SM system. The traceability definition is described by a security game between an adversary and a challenger. Let *q′* be the total number of key generation queries performed by the adversary *A*, and the game between challenger *C* and *A* is as follows:

• Setup: *C* calls **Setup**(1*k*) algorithm and returns the
public parameters *PK* to *A*..

Definition 2. The modified ABKS-SM system is fully traceable if there exists no polynomial time *A* that has a nonnegligible advantage in breaking the above game.

Security Requirements

Similar to security requirements in typical private information retrieval schemes both the basic and modified ABKS-SM systems should ensure the following privacy requirements:

• **Data privacy**. **DU**s can access the shared data, if and only if, they have valid authorization from multiple **DO**s.

• **Privacy for DUs**. **CSP** is convinced that **DU**'s search queries are authorized by **DO**s, without learning any potential information about the queried content.

• **Privacy for DOs**. Even if a part of **DO**s is corrupted, the adversary cannot forge valid authorizations from remaining **DO**s as there exist no interactions and additional computing operations among multiple **DO**s. As will be shown in **Theorem** 3 in Section 6.1, the basic and modified ABKS-SM systems satisfy the above privacy requirements if they achieve selective security in the generic bilinear group model.

VIII. PROPOSED ABKS-SM SYSTEMS

In this section, we first present the concrete construction of the basic ABKS-SM system, which supports fine-grained keyword search and hidden access policy. Then, we explain how the basic ABKS-SM system is extended to achieve malicious user tracing in the modified ABKS-SM system.

Construction of Basic ABKS-SM Syetem

Unlike existing CP-ABKS schemes, we consider a *shared* multi-owner setting where each file is co-owned by a group of **DO**s. In the basic ABKS-SM system, we use conventional symmetric encryption algorithm (AES, DES, etc.), access matrix$Md \times l$ (or (d, l)-LSSS), and access policy P, to respectively encrypt files, file encryption keys and keywords. Even though a certain **DU** can issue search queries and obtain the returned search results, he/she cannot decrypt the encrypted data without valid authorizations from multiple **DO**s. Moreover, in practice, the access policies contain sensitive information and should also be protected. However, existing CP-ABKS schemes with hidden access policies are not practical since any malicious **DU** having the same attribute set with others, can leak his/her decryption privilege without fear of being caught. Thus, we further extend the traceability function in the modified ABKS-SM system, and present a concrete construction. Note that we design a two-level access control over outsourced files. As for the firstlevel access control over file decryption,we design an access matrix$Md \times l$, which is used to encrypt each file encryption key according to **DU**'s identity list by leveraging LSSS.

Security Analysis

First, based on the aforementioned generic bilinear map model, the basic and modified ABKS-SM systems can achieve selective security. Second, the modified ABKS-SM system can achieve the full traceability under ϕ-SDH assumption. Finally, the privacy protection (including privacy for data, **DU**s and **DO**s) can be also achieved under DBDH assumption in the basic and modified ABKS-SM systems.

IX. CONCLUSION

In the paper, we presented a practical attribute-based keyword search scheme supporting hidden access policy in the *shared* multi-owner setting. Furthermore, we demonstrated how the basic ABKS-SM system can be extended to support traceability (i.e., tracing of malicious **DU**s) in the modified ABKS-SM system, if desired. The formal security analysis showed that the basic and modified ABKS-SM systems achieve selective security and resist off-line keyword guessing attack in the generic bilinear group model. We also demonstrated the utility of the proposed ABKS-SM systems by evaluating their performance using three real-world datasets and on a testbed including 11 mobile terminals and a high performance workstation server. One limitation of the proposed ABKS-SM systems is that as the number of system attributes increases, so does the computational and storage costs. Thus, we intend to improve the efficiency of the ABKS-SM systems in the future. Also, to facilitate the efficient locating of search results and minimizing the number of irrelevant search results, we will

focus on expressive search (e.g., multi-keyword search and fuzzy keyword search) in our future work.

REFERENCES

[1] J. Li, W. Yao, Y. Zhang, H.Qian,and J. Han, "Flexible and finegrained attribute-based data storage in cloud computing," *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 785–796, 2017.

[2] J. Li, H. Yan, and Y. Zhang, "Certificateless public integrity checking of group shared data on cloud storage," *IEEE Transactions on Services Computing*, vol. PP, pp. 1–1, 2018.

[3] D. Wu, J. Yan, H. Wang, D. Wu, and R. Wang, "Social attribute aware incentive mechanism for device-to-device video distribution," *IEEE Transactions on Multimedia*, vol. 19, no. 8, pp. 1908–1920

[4] [4] D. Wu, Q. Liu, H. Wang, D. Wu, and R. Wang, "Socially aware energy efficient mobile edge collaboration for video distribution," *IEEE Transactions on Multimedia*, vol. 19, no. 10, pp. 2197–2209, 2017.

[5] D. Wu, S. Si, S. Wu, and R. Wang, "Dynamic trust relationships aware data privacy protection in mobile crowd-sensing," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2017.

[6] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proc. IEEE Symposium on Security and Privacy (SP 2000)*, 2000, pp. 44–55.

[7] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proc. International conference on the theory and applications of cryptographic techniques (EUROCRYPT 2004)*, 2004, pp. 506–522.

[8] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 3, pp. 312–325,2016.

[9] R. Chen, Y. Mu, G. Yang, F. Guo, and X. Wang, "Dual-server public-key encryption with keyword search for secure cloud storage," *IEEE transactions on information forensics and security*, vol. 11, no. 4, pp. 789–798, 2016.

[10] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symposium on Security and Privacy (SP 2007)*, 2007, pp. 321–334.