An Efficient Multi Owner Data Sharing Using Attribute Based Encryption Data Retrieval in Cloud

[1]Shaik Kalim Peerulla Basha & [2]A. Chandra Obula Reddy

[1]M-Tech, Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences.

[2]Assistant Professor, Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences, Kadapa-Andhra Pradesh.

ABSTRACT:

Pertinently a great attention to detailed access control and the process of searching queries were made easier over encrypted data in the cloud by the Cipher text-Policy attribute based keyword search (CP-ABKS). Unfortunately, CP-ABKS strategies were designed to support unshared multi-owner setting. In fact, by without incurring high computational and storage costs CP-ABKS cannot be applied in the shared multi-owner setting, because in the shared multi-owner setting the record will be accredited by a fixed number of data owners. Some other privacy concerns on access policies; mostly existing schemes are vulnerable to offline keyword guessing attacks if the keyword space is of polynomial size. When more than one data user has the same subset of attributes it becomes even more difficult to identify malicious users who leak the secret keys. Here we represent a privacy-preserving CP-ABKS system with hidden access policy in Shared Multi-owner setting (basic ABKS-SM system), and demonstrate how it's improved to support malicious user tracing (modified ABKS-SM system). Later we prove that the proposed ABKS-SM systems achieve selective security and resist off-line keyword-guessing attack in the generic bilinear group model and evaluate their working process using real-world datasets to show its feasibility and practicality in a broad range of actual scenarios by without incurring additional computational burden.

KEYWORDS: Cipher text-Policy attribute based keyword search (CP-ABKS)

1. INTRODUCTION

Cloud computing is a flexible solution that allows hospitals to leverage a network of remotely accessible servers where they can store large volumes of data in a secure environment that is maintained by IT professionals. Since the introduction of the EMR mandate, health care organizations across the United States have adopted cloud computing solutions as a means of storing and protecting patient EMRs. The federal mandate for electronic medical records which took effect on January 1st, 2014, was signed into law as part of the American Recovery and Reinvestment Act. The mandate requires hospitals and health care facilities to demonstrate meaningful use of electronic medical records for storing information about patient interactions. The stated goals of meaningful use are to improve the quality, safety and efficiency of medical services, to engage patients and family, improve the coordination of care, and to maintain patient privacy and security. The implementation of cloud storage for electronic medical records has streamlined the process of collaborative patient care in America. Cloud-based medical records have made it easier for doctors to collaboratively view or share a patient's medical records. In the past, a patient might have a separate file for medical records at each doctor they visit – some records at their family doctor, some kept by a dentist, some at one specialist's office and some at another.

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 (rather than one-to-one) 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 (rather than one-to-one) (CP-ABKS). However, in many applications, data records are co-owned by a number of data owners, rather than a single data owner. 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 in Fig.1

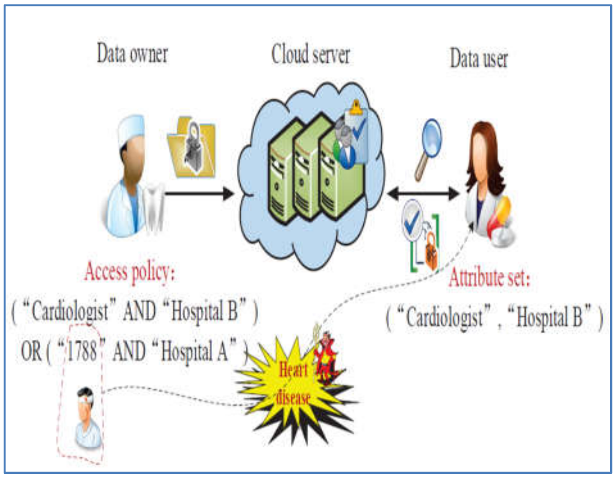
Data owner
Cloud server
Data user
Access policy:
("Cardiologist" AND "Hospital B")
OR ("1788" AND "Hospital A")
Attribute set:
("Cardiologist" , "Hospital B")
Heart disease

Fig.1.1: Privacy leakage in access policy.

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") – see Fig. 1. 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 juris- dictions. Hence, there have been efforts to design CP-ABE scheme with hidden access policies.

There have also been efforts to design schemes that allow a data owner to delegate his/her search capability in a fine- grained manner, which allows other data users to search, retrieve and decrypt encrypted data of interest. Examples include Ciphertext-Policy Attribute-Based KeywordSearch (CP-ABKS) . However, in manyapplications, 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 [15], [16] in the unshared multi-owner setting(wheremultipledataownersmanagedifferentdata 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 multi- owner setting and shared multi-owner setting are described in Fig.2.

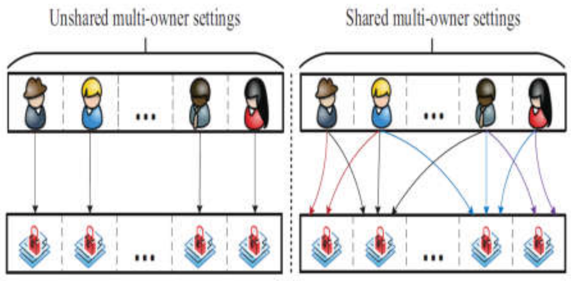

Fig. 2. Differences between *unshared* and *shared* multi-owner setting.

Most CP-ABKS schemes do not consider the case where dishonest data users may share their secret keys with unauthorized entities, resulting in unauthorized entities having the same privileges as dishonest data users. Thus, it is necessary to support traceability in CP-ABKS schemes, in order to trace malicious data users who, sell or leak their secret keys [20].

At the time of this research, there is no practical CP- ABKS system that supports hidden access policy and traceability simultaneously in shared multi-owner setting. Hence,inthispaperwe first propose a privacy-preserving Attribute-Based Keyword Search system with hidden access policy in Shared Multi-Owner setting (basic ABKS-SM system), then extend this basic system to support traceability (modified ABKS-SM system). Specifically, the main contributions of this paper are as follows.

- Sharedmulti-ownersetting.BothABKS-SMsystems 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 privilegedrecipients.
- Tracing of malicious data users. To prevent dis- honest data users from leakingtheir secret keys to others (e.g., for profits), the modified ABKS-SM system provides traceability by securely embedding their identity information in the secretkeys.

We formally prove that the basic andmodified ABKS- SM systems guarantee the security of shared data and access policies, achieve selective security, and resist off-line keyword-guessing attack in the generic bilinear groupmodel. We also demonstrate performanceof the basic ABKS-SM system using experiments on real-worlddatasets.

II LITERATURESURVEY

The first symmetric SE scheme and asymmetrical SE scheme were presented by Song et al. [6] and Boneh et al. [7], respectively. Subsequent SE schemes were designed to support arange of features, such as single keyword search [2], multi-keyword search [8] and ranked keyword search [2], [4]. 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,Zheng et al. [5] presented the CP-ABKS scheme that enables data owners to grant fine-grained search permissions, Sun et al. [1] 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 insuch schemes, Li et al. [6] implemented a keyword search function in attribute-based encryption (ABE) scheme, by outsourcing key-issuing and decryption operations. Dong et al. [7] 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 theaccess policy embedded in the ciphertexts may leak sensitive information to authorized data users, as discussed in the preceding section. Thus, Nishide et al. [2] constructed a more practical CP-ABE scheme, which allows the encryptor to use wildcards to represent certain attributes in a hidden solution. Similarly, Phuong et al. [4] proposed 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, Qiu et al. presented 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. [3] provided privacy-preserving ranked multi-keyword search in the multi-owner model and prevented attackers from eavesdropping secret keys. Miao et al. [5] 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.Another limitation of CP-ABKS schemes is that an honest-but-curious cloud service provider may seek to learn additional sensitive information, other than the stored ciphertexts and submitted trapdoors. Also secret keys (or decryption keys) are defined over different attribute sets, rather than their corresponding identities. Hence, while CPABKS schemes can achieve one-to-many encryption and support expressive access control, they are not capable of identifying data users leaking the secret keys if the 'culprits' have the same subset of attributes as other honest data users. Hence, a data user may choose to deliberately trade his/her (partial or entire) decryption privileges for profit without being caught. Thus, traceability should be incorporated in the design of CP-ABKS schemes to facilitate accountability. Based on the traceable CP-ABE technique [3], we extend the traceability feature in the basic ABKS-SM system to construct the modified ABKS-SM system so that the requirements of real-world applications can be satisfied.

III. EXISTING SYSTEM

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, Zheng et al. 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, Li et al. 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, Nishide et al. constructed a more practical CP-ABE scheme, which allows the encryptor to use wildcards to represent certain attributes in a hidden solution. Similarly, Phuong et al. proposed 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.

Disadvantages

- In the existing work, while CPABKS schemes can achieve one-to-many encryption and support expressive access control, they are not capable of identifying data users leaking the secret keys if the ‘culprits’ have the same subset of attributes as other honest data users.
- The existing system has one serious limitation of CP-ABE schemes which is that the access policy embedded in the cipher texts may leak sensitive information to authorized data users, as discussed in the preceding section.

IV. PROPOSED SYSTEM

In this system the system first proposes a privacy-preserving Attribute-Based Keyword Search system with hidden access policy in Shared Multi-owner setting (basic ABKS-SM system), then extend this basic system to support traceability (modified ABKS-SM system). Specifically, the main contributions of this paper are as follows.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 cipher texts 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.

Advantages

- The system is more effective since Linear Secret Sharing Schemes (LSSS) present to give more security on the system.
- The system is more secured since 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.

V. MODULES

Healthcare Service Provider

In this module, Provider has to register to cloud and View all the CDA received and request to the cloud to access the generated CDA from hospital - A & hospital - B. Once the access request is granted by the cloud the provider will write the reply letter for corresponding CDA reports and sends.

Patient/End User

In this module, the user/patient Registers to cloud and is authorized by the cloud and Logs in the user/ patient has to request the search key to search the patient CDA and also request for the view permission from the cloud. If the permission is provided by the cloud the corresponding user/patient can view the CDA generated and the corresponding reply from the doctor.

Hospital - A

In this module, CDA is generated, encrypted as hospital-A document and then uploaded to cloud, also can view the CDA replies from Healthcare service provider and can view all the generated CDA’s.

Hospital - B

In this module, CDA is generated, encrypted as hospital-B document and then uploaded to cloud, also can view the CDA replies from Healthcare service provider and can view all the generated CDA’s.

Cloud Server

In this module the cloud will authorize both the doctor and the patient/user. Receive all CDA generated from the hospitals and store, Select the doctor and Sends the CDA report for corresponding doctor. Provide permission for the CDA requests requested by the provider and also generates the search key requested by the user. This module shows the charts/Results based on the CDA.

CONCLUSION

The CDA document format a clinical information standard designed to guarantee interoperability between hospitals, a large number of HIE projects that use the CDA document format have been undertaken in many countries. Table 5 shows various HIE projects and whether they generate CDA documents or integrate multiple CDA documents. Our cloud computing based CDA generation and integration system has a few pronounced advantages over other existing projects. First, hospitals do not have to purchase propriety software to generate and integrate CDA documents and bear the cost as before. Second, our service is readily applicable to various developer platforms because an Open API is to drive our CDA document generation and integration system. Regardless of the type of the platform, CDA documents can be easily generated to support interoperability. Third, CDA document generation and integration system based on cloud server is more useful over existing services for CDA document.

REFERENCES

[1] *Y. Kwak "International standards for building electronic health record (ehr)" <em>Proc. Enterprise Netw. Comput. Healthcare Ind.</em> pp. 18-23 Jun. 2005.*

[2] M. Eichelberg T. Aden J. Riesmeier A. Dogac Laleci "A survey and analysis of electronic healthcare record standards" <em>ACM Comput. Surv.</em> vol. 37 no. 4 pp. 277-315 2005.

[3] T. Benson Principles of Health Interoperability HL7 and SNOMED 2009 Spinger.

[4] J. Lähteenmäki J. Leppänen H. Kaijanranta "Interoperability of personal health records" <em>Proc. IEEE 31st Annu. Int. Conf. Eng. Med. Biol. Soc.</em> pp. 1726-1729 2009.

[5] R. H. Dolin L. Alschuler C. Beebe "The HL7 Clinical Document Architecture" <em>J. Am. Med. Inform. Assoc.</em> vol. 8 pp. 552-569 2001.

[6] R. H. Dolin L. Alschuler S. Boyer "The HL7 Clinical Document Architecture" <em>J. Am. Med. Inform. Assoc.</em> vol. 13 no. 1 pp. 30-39 2006.

[7] M. L. Müller F. Ǖckert T. Bürkle "Cross-institutional data exchange using the clinical document architecture (CDA)" <em>Int. J. Med. Inform.</em> vol. 74 pp. 245-256 2005.

[8] H. Yong G. Jinqiu Y. Ohta "A prototype model using clinical document architecture (cda) with a japanese local standard: designing and implementing a referral letter system" <em>Acta Med Okayama</em> vol. 62 pp. 15-20 2008.

[9] K. Huang S. Hsieh Y. Chang "Application of portable cda for secure clinical-document exchange" <em>J. Med. Syst.</em> vol. 34 no. 4 pp. 531-539 2010.

[10] https://ijret.org/volumes/2015v04/i06/IJRET20150406028.pdf

[11] https://www.researchgate.net/publication/339687406_HS2Cloud_A_Secure_Lightweight_Framework_for_Image_Storage_on_Hybrid_Cloud/citation/download.