Using UBE to Share the Private Medical Record in Cloud Server

R. Navaneethakrishnan[1] , S. Parameswaran[2]

[1]*CSE Department PG scholar*
Sree Sowdambika College of Engineering
[2]*CSE Department Asst.Professor*
Sree Sowdambika College of Engineering

Abstract

Private Medical Records (PMR) is an emerging patient medical data exchange. The PMR is stored in third party semi trusted server like could service. Now we face wide secrecy worries as medical data could be exposed to those semi trusted servers and to unlicensed access. To assure the patients' control over access to their own PMRs, it is a hopeful method to encrypt the PMRs before outsourcing. Here some risks of secrecy coverage, key management, smooth access, and ability of user reversal, have remained the most significant tasks toward achieving accurate, cryptographically data access control in cloud servers. In this paper, we recommend a new patient-centric framework and a suite of mechanism for data access control to PMRs stored in semi trusted servers. We reach accurate and scalable data access control for PMRs, we force user-based encryption (UBE) mechanism to encrypt each patient's PMR file. we focus on the several data vender state, and split the users in the PMR system into multiple security domains that significantly reduces the key management complexity for users. A high degree of patient secrecy is assured concurrently by using multi authority UBE. The system also supports lively change of access policies or file, supports efficient on-demand user reversal and break-glass access under critical situations. Our analytical and experimental results are presented which show the safety, scalability and efficiently of our proposed system.

Index Terms— *private medical records, cloud computing, data secrecy, accurate access control, user-based encryption.*

1. Introduction

We store the PMR in cloud semi trusted server. A PMR service allows a patient to create and manipulate his health data in one place over the network, which is used to storing, retrieval, and transfer of the medical data more efficient. In this model each patient is allowed to control access rights as his medical information and can share his health information with a wide range of users. Including Insurance Company, relatives. Alternative of construct and maintaining dedicated data centers which more cost, third party semi trusted servers can be used to store the PMR. Successful examples are, Microsoft Azure [4] Google App Engine [3] Amazon's EC2 and S3 [2], and which provide users with scalable resources in the pay-per-use use method at relatively low cost. One of the major problems raised by data outsourcing is confidentiality. Data confidentiality is not only a secrecy issue, but also of juristic concerns. In patient management application situation use and disclosure of protected medical information (PMI) should meet the requirements of Health Insurance Portability and Accountability Act (HIPAA),[6] and keeping user data confidential against the storage servers is requirement. The high value of the sensitive personal Medical information (PMI), the semi trusted storage servers are often the targets of various malicious actions which may lead to exposure of the PMI. To guarantee patient-centric privacy control over their own PMRs, it is basic to

have accurate data access control techniques that work with third party servers. To deal with the risks of secrecy exposure, instead of letting the PMR service providers encrypt patients' data, PMR services should give patients full control over the selective sharing of their own PMR data. To this end, the PMR data should be encrypted in addition to traditional access control techniques access control techniques provided by the system [7]. A PMR file should only be available to the users who are given the equivalent decryption key, while remain personal to the rest of users. Due to the high cost of developing and preserving personal data servers, many PMR services are provided by cloud service providers, there are many security risks which could impede its wide espousal. The main alarm is whether the patients could really control the share the their sensitive private medical information (PMI), especially when they are stored on a semi trusted server which publics may not fully trust. Cloud servers are generally not covered things. Due to the high value of the complex private medical information (PMI), the semi trusted storage servers are often the targets of various attacks which may lead to reveal the PMI. A feasible method would be to encode the PMR before outsourcing. The PMR owner himself should decide how to encode his records and to allow which set of users to access to each record. A PMR file should only visible to the users who are provide the equivalent decryption key, on the other hand, the patient right to not only grant, but also cancel access rights when they sense it is required. The aim of patient-centric secrecy is often in conflict with scalability in a PMR system. The authorized users may either need to access the PMR for own use or professional purposes. Examples are relatives, friends then medical doctors, pharmacists, etc.. we refer to the two categories of users as private and skilled users, respectively. The latter has potentially large scale; should each owner himself be directly responsible for managing all the skilled users, he will easily be increased by the key management problem. This type of users' access requests are normally unpredictable, it is difficult for an owner to regulate a list of them. At the same time, various from the single data owner situation considered in most of the existing works in a PMR system, there are multiple owners who may encode allowing to their own methods, possibly using various sets of cryptographic keys. Each user obtain keys from every owner whose PMR he wants to read accessibility since patients are not always connected in the network. An another way is to provide a central authority (CA) to do the key management on replace the all PMR owners, but this requires too much trust on a single authority. In this paper, we try to study the patient-centric, safe sharing of PMRs stored on semi trusted servers, and focus on addressing the difficult and challenging key management issues. In order to protect the personal medical information stored on a semi trusted server, we adopt user based encryption (UBE) as the main encryption primitive. Using UBE, access policies are expressed based on the users or data, which allows a patient to selectively share his PMR among a set of users by encoding the record under a set of conditions, without the need to know a complete list of users. The difficulties per encoding, key generation, and decoding are only linear with the number of users involved. However, to participate UBE into a large-scale PMR system, significant issues such as key organizing, dynamic policy updates, and effective on-demand reversal are nontrivial to solve. We make the following main contributions:

1. We propose a new UBE-based architecture for patient-centric safe sharing of PMRs in cloud server systems, under the multi owner scenarios. To address the key organizing problems, we divide the users in two types of domains, namely open and private domains. In our system can concurrently handle different types of PMR sharing applications' requirements, while incurring minimal key organizing over head for both owner sand users in the system. In addition, the architecture applies write access control, handles dynamic strategy updates, and provides break-glass access to PMRs under emergence scenarios.

2. In the open domain, we use multi authority UBE (MA-UBE) to improve the safety and avoid key escrow problem. Each user authority (UA) in it governs a disjoint subset of user policies, while no one can able to control the security of the whole system. We suggest techniques for key sharing and encryption so that PMR owners can policies during file encryption. In the private domain, owners

directly assign access privileges for private users and encrypt a PMR file under its data attributes. Furthermore, we enhance MA-UBE by putting forward an efficient and on-demand user revocation arrangement, and prove its security under standard safety assumptions. Inpatients have full secrecy control over their PMRs.

3. We provide a thorough analysis of the difficulty and safety of our proposed secure PMR sharing solution, in terms of multiple metrics in calculation, record communication, storage, key management and user based download. We also compare our scheme to several previous ones in complexity and safety. We reveal the efficiency of our system by implementing it on a modern workstation and performing experiments. Compared with the initial version of this paper [1], there are some main additional contributions:

1. We explain and extend our usage of MA-UBE in the open domain, and properly show how and which types of users-defined record access policies are realized.

2. We clarify the proposed revocable MA-UBE scheme, and provide a formal security proof for it.

3. We carry out both real-time experiments to evaluate the performance of the proposed solution in this paper.

2. Related Work

Traditional Access Control for EMARS

Traditional access control in electronic medical records (EHRs) often places full trust on the health care server where the EMR data are often resided in, and the access policies are applied and enforced by the health servers. Different access control models have been projected and applied, with role-based (RBAC) and attribute-based access control (ABAC). The RBAC, each user's access right is allocated based on his roles the ABAC derived the role concept in RBAC to attributes, some properties of the resource, entities, and the environment. Matched with RBAC, the ABAC is more efficient in the context of health care due to its potential flexibility in strategy descriptions. A line of this paper aims at increase the effectiveness and flexibility of the access control policies. For personal health records (PHRs) in cloud environments, the PHR service providers may be in the semi trust domains with the patients'. Thus patient-centric secrecy is difficult to guarantee when full trust is placed on the cloud servers this time the patients lose physical control to their sensitive data. The PHR needs to be encrypted in a way that enforces each patient's personalized secrecy policy, which is the consideration of this paper.

Cryptographic Enforced Access Control for Semi Trusted Servers

For access control of outsourced data, semi trusted servers are often assumed. With cryptographic methods, the goal is trying to apply that who has access to which parts of a patient's PHR data in a suitable way. Symmetric key cryptographic (SKC) based solutions. Vimercati et.al. Proposed a way for securing outsourced data on third party servers based on symmetric key methods, which can reach accurate access control. On the other hand, the convolutions of record creation and user acceptation/revocation operations are linear to the number of authorized users, which is less scalable. Files in a PHR are arranged by hierarchical groupings in order to make key distribution more flexible. In SKC-based solutions have some key limitations. First, the key management overhead is high when there are a more number of users and owners, key distribution can be very difficult when there are multiple owners, and it requires each owner to always be online. Second, user revocation is ineffective, upon revocation of one user, all other users will be affected and the data need to be re-encrypted. Furthermore, user's read and writes rights are not separate. Public key cryptography (PKC) based solutions. PKC based solutions were projected due to its ability to separate read and write privileges. Benaloh et.al. Proposed a structure based on hierarchical identity based encryption (HIBE), still has possibly high key management overhead. In order to deal with the multi-user situations in encrypted search, Dong et.al. projected a solution based on proxy encoding. Access control can be imposed if every read and write operation involve a proxy server. It is also does not maintenance flexible access control, and is not safe. User-based Encryption (UBE). The SKC and PKC based solutions all arise low scalability in a large PMR

system, record encryption is done in an one-to-one way, while each PMR may have large number of users. To avoid such problems, new one-to-many encryption methods such as user-based encryption can be used. In the seminal paper on UBE, data is encrypted to a group of uses characterized by a set of attributes, which theoretically makes the key management more effective. Then, several works used UBE to realize fine-grained access control for outsourced PMR. However, they have not addressed the multiple owner settings, and there some fault in framework for patient-centric PMR access control in multi-owner PMR systems. In a single authority for all users and owners is adopted. This affected from the key escrow problem, and patients' secrecy still cannot be sure the authority has keys for all owners. Recently Ibraimi et.al. applied cipher text policy ABE (CP-ABE) to manage the sharing of PMRs. They still adopt a single public authority, while the challenging key-management issues.

In old ABE-based framework for patient-centric safe sharing of PMRs in cloud server environments, in the multi-owner settings. To address the key management problems, we divide the users into two types, namely open and private domains. In the majority medical users are managed distributive by user authorities in the former, each PMR owner manage the keys of a minimum number of users in his private domain.

In the open domain, we use multi-authority ABE (MA-ABE) to improve the security and avoid key escrow problem. The user authority (UA) manages a disjoint subset of user policies, but none of them alone is able to manage the security of the whole system mechanisms for key delivery and encryption so that PMR owners can specify personalized fine-grained role-based access policies during record encryption. In the private domain, owners directly assign access privileges for private users and encrypt a PMR file under its data attributes.

3. Proposed System

Public Domain Approach

In existing system, public users have to getting key from user authority to access records. Every time public users need to do this process. It takes long time to do. Public users do not like to spend lot of time for one process.

In proposed system, user authority allocates some set of secret question to each public user. The questions chosen by the public users. If public user gives correct answer to the secret question, then the public key is generated for the particular user. By using that public key, the record is accessed by the public user. It avoids the user authority in busy state and also provides safe access the PMR. In this process we increase the user access in efficient way to improve the scalability.

Improved Break-Glass Access

In previous paper certain parts of the PMR data, medical employees need to have temporary access when an emergency happens to a patient, who may become insensible and is unable to change his access policies beforehand. The medical employees will need some temporary access policies (e.g., emergency key) to decrypt those data. Under our system, this can be normally reach by letting each patient share his emergency key to an emergency department. Purposely, in the beginning, each owner defines an "emergency" attribute and builds it into the private part of the cipher text of each PMR document that he allows break-glass access. Then he generates an emergency key skEM using the single node key-policy "emergency", and delegates it to the emergency department who keeps it in a database of patient file. Upon emergency, a medical employees authenticates himself to the emergency department, requests and obtain the corresponding patient's skEM, and them decrypts the PMR files using skEM. After the patient recovers from the emergency condition, he can revoke the break-glass access by computing a rekey: rkEM, submit it to the emergency department and the server to update his skEM.

In above work, the key revocation is done while the user informs the server after discharge from the hospital. Because of that, the information about the patient is still available and there is a chance for misuse that information. But now, there is a time limit for the key validation. Key

validation time may or may not be expire while the patient is in emergency ward or not. After key expire one emergency message will send to family doctor and friend with the help of that key validation, we can protect the information from the access of unauthorized person. These information are available based on the privileges of the particular person.

Encryption and Decryption Approach

Encryption is the process of changing data into a form that can be read only by the intended receiver. To decipher the message, the receiver of the encrypted data must have the proper decryption key.

Advanced **E**ncryption **S**tandard, a symmetric 128-bit block data encryption technique. Replacing the DES encryption it used. AES works at multiple network layers simultaneously.

The AES replaced the DES with new and updated features:

- Block encryption implementation
- 128-bit group encryption with 128, 192 and 256-bit key lengths
- Symmetric algorithm requiring only one encryption and decryption key
- Data security for 20-30 years
- Worldwide access
- No royalties
- Easy overall implementation

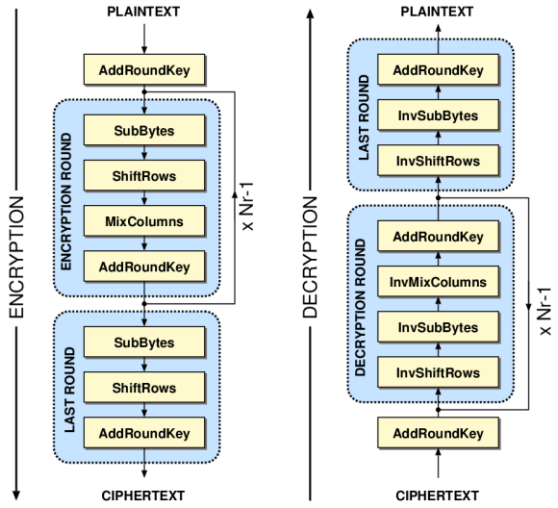

Fig.1 AES Encryption and Decryption

Key Generation

1. Generate p and q two large prime numbers

2. Let n= pq

3. Let m= (p-1) (q-1)

4. Choose a small number e, co prime to m

5. Find d, such that de% m= 1.

Publish e and n as the public key. d and n as the secret key.

Encryption

$C=P_e \% n$

Decryption

$P=C_d \% n$

x % y means the remainder of x divided by y this secret key is distributed to the data access members through email address. The email address of the data access member will be known to PMR owners in advance only. Using this secret key data access member can access the files required.

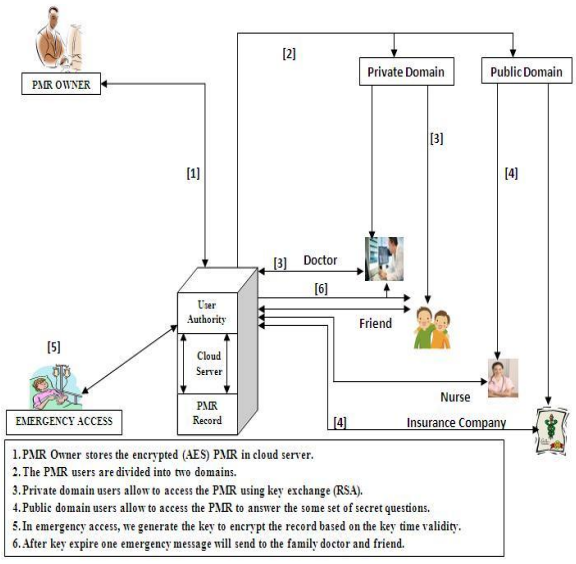

Fig.1 Architecture for PMR Cloud System

Table.1 User Access Table

User/ Access Type	Doctor	Friend	Nurse	Insurance Company
Read	Yes	Yes	Yes	Yes
Write	Yes	No	No	No
Download Type	Word	PDF	PDF	Image

Goal

The main objective of patient-based secrecy is often in disagreement with scalability in a PMR system. The allowed users may either need to access the PMR for personal use or medical purposes. Examples of the former are relatives and friends, while the latter can be medical doctors, and insurance company etc. We refer to the two categories of users as private and medical users, correspondingly.

On-demand user revocation facility

If a user does not access their account for a particular period of time, then the account will blocked automatically. The PMR user tries to access the record with wrong security key in more than 3 times, then the user's privileges will be revoked. PMR owners have the rights to revoke the user's privileges in dynamically.

Benefits in proposed system

- There is policy management for PMR access, data access member can able to access the records which they have rights set by the policy.
- PMR stored in semi trusted cloud server are in encrypted form and no one can view and change the content without authorization.
- There is a structured way to access the record for private and professional purpose through user access policies and user based encryption and decryption.
- Dynamic on demand user revocation
- Improve the emergency access methods.

3. Conclusion

In this paper, we have proposed a new architecture of safe sharing of personal medical records in cloud server. Considering semi trusted cloud servers, we argue that to fully understand the patient-centric concept, patients can have full control of their own secrecy through encrypting their PMR files to allow flexible access. The system addresses the exclusive challenges brought by multiple PMR owners and users, in that we reduce the difficulty of key management while enhance the secrecy guarantees compared with existing works. We utilize UBE to encrypt the PMR data, so that patients can allow access not only by private users, but also various users from open domains with different professional roles, experience and affiliations.

REFERENCES

[1] Scalable and secure sharing of Personal Health Records in Cloud Computing using Attribute based encryption- Ming Li Member, IEEE, Shucheng Yu, Member, IEEE, Yao Zheng, Student Member, IEEE, Kui Ren, Senior Member, IEEE, and Wenjing Lou, Senior Member, IEEE.

[2]H. L öhr, A.-R. Sadeghi, and M. Winandy, "Securing the e-health cloud," in Proceedings of the 1st ACM International Health Informatics Symposium, ser. IHI '10, 2010, pp. 220-229

[3]M. Li, S. Yu, N. Cao, and W. Lou, "Authorized private keyword search over encrypted personal health records in cloud computing," in ICDCS '11, Jun.2011

[4]S. Yu, C. Wang, K. Ren, and W. Lou, "Achiev secure, scalable,and fine-grained data access control in cloud computing," in IEEE INFOCOM'10, 2010.

[5]C.Dong, G.Russello, and N.Dulay, "Shared and Search a encrypted and for untrusted servers," in Journal of Computer Security,2010.

[6] "Google, Microsoft say hipaa stimulus rule doesn't apply to them," http://www.ihealthbeat.org/Articles/2009/4/8/.

[7]"The Health insurance portability and accountability act.