IJDCST @Feb-March-2015, Issue- V-3, I-3, SW-12
ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

Privacy Preserving and Secure Personal Health Record (PHR) management under Multi-owner Settings

PGK Kiran Kumar, Lalu Nayak

Abstract: With the emergence of cloud computing, it is attractive for the personal health record (PHR) service providers to shift their PHR applications and storage into the cloud, in order to enjoy the elastic resources and reduce the operational cost. However, by storing PHRs in the cloud, the patients lose physical control to their personal health data, which makes it necessary for each patient to encrypt her PHR data before uploading to the cloud servers. In particular, the data has to be stored on a central server locked by the access control mechanism, and the data owner loses control on the data from the moment when the data is sent to the requester. Therefore, these mechanisms do not fulfill the requirements of data outsourcing scenarios where the third party storing the data should not have access to the plain data, and it is not trusted to enforce access control policies. In this paper, we describe a new approach which enables privacy preserving secure storage and controlled sharing of patient's health records in the aforementioned scenarios. A new variant of a ciphertext-policy attribute-based encryption scheme is proposed to enforce patient/organizational access control policies such that everyone can download the encrypted data but only authorized users from the social domain (e.g. family, friends, or fellow patients) or authorized users from the professional domain (e.g. doctors or nurses) are allowed to decrypt it. We achieve this goal by exploiting and uniquely combining techniques of privacy preserving attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. Our proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability under Multi-owner Settings. Extensive analysis shows that our proposed scheme is highly efficient and provably secures under existing security models.

Keywords: PHRs, CP-ABE, Privacy Preserving Access, secure cloud storage, Multi-owner settings.

I. Introduction

Cloud computing is a promising computing paradigm which recently has drawn extensive attention from both academia and industry. At the same time, cloud computing has attracted a lot of attention because it provides storage-as-a-service and software-as-a-service, by which software service providers can enjoy the virtually infinite and elastic storage and computing resources [1]. As such, the PHR providers are more and more willing to shift their PHR storage and application services into the cloud instead of building specialized data centers, in order to lower their operational cost. For example, two major cloud platform providers, Google and Microsoft are both providing their PHR services, Google Health and Microsoft HealthVault, respectively.

While it is exciting to have PHR services in the cloud for everyone, there are many security and privacy risks which could impede its wide adoption. The main concern is about the privacy of patients' personal health data and who could gain access to the PHRs when they are stored in a cloud server. Since patients lose physical control to their own personal health data, directly placing those sensitive data under the control of the servers cannot provide strong privacy assurance [2, 3 and 4] at all. First, the PHR data could be leaked if an insider in the cloud provider's organization misbehaves, due to the high value of the sensitive personal health information (PHI). To deal with the potential risks of privacy exposure, instead of letting the PHR service providers encrypt patients' data, PHR services should give patients (PHR owners) full control over the selective sharing of their own PHR data. To this end, the PHR data should be encrypted in addition to traditional access control mechanisms provided by the server [4]. Basically, each patient shall generate her own decryption keys and distribute them to her authorized users. In particular, they shall be able to choose in a fine-grained way which users can have access to which parts of their PHR; for the unauthorized parties who do not have the corresponding keys, the PHR data should remain confidential.

On the other hand, since there are multiple owners, each user may have to obtain keys from every owner whose PHR he wants to read, limiting the accessibility since not every patient will be always online. Yet, in a straightforward solution where all the users are managed by some central authority (CA) instead of each owner, the CA will have the ability to decrypt all the owners' data, such that owners have no full control over their data and their privacy will still be at risk. While various previous works proposed techniques for cryptographically enforced access control to outsourced data [4, 6, 7, 8 and 9], they focused on single-owner architecture which cannot directly solve the above challenges under multi-owner scenario in PHR system. Therefore, a new framework for patient-centric access control suitable for multi-owner PHR systems is necessary.

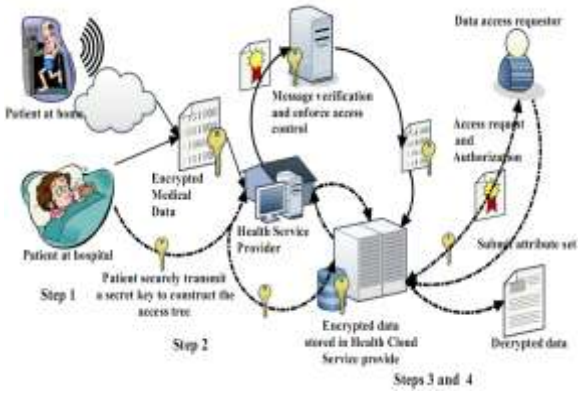

Figure 1. Privacy Preserving and Secure PHR management

In this paper, we describe a new approach which enables privacy preserving secure storage and controlled sharing of patient's health records in the aforementioned scenarios as shown in figure 1. A new variant of a ciphertext-policy attribute-based encryption scheme is proposed to enforce patient/organizational access control policies such that everyone can download the encrypted data but only authorized users from the social domain (e.g. family, friends, or fellow patients) or authorized users from the professional domain (e.g. doctors or nurses) are allowed to decrypt it. We achieve this goal by

exploiting and uniquely combining techniques of privacy preserving attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. Our proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability under Multi-owner Settings. Extensive analysis shows that our proposed scheme is highly efficient and provably secures under existing security models.

II. Literature Review

In an electronic health record system, patients, healthcare providers, and medical devices can upload health records and retrieve and view them at a later time. Furthermore, patients may delegate access rights and allow family, friends, and designated healthcare providers to view or to edit parts of their record. Patients and their delegates may wish to efficiently perform searches in a unique manner over part or all of the record.

Evolution of PHRs: Traditionally, research on access control in electronic health records (EHRs) often places full trust on the health care providers where the EHR data are often resided in, and the access policies are implemented and enforced by the health providers. Various access control models have been proposed and applied, including role-based (RBAC) and attribute-based access control (ABAC) [10]. In RBAC [11], each user's access right is determined based on his/her roles and the role-specific privileges associated with them. The ABAC extends the role concept in RBAC to attributes, such as properties of the resource, entities, and the environment. Compared with RBAC, the ABAC is more favorable in the context of health care due to its potential flexibility in policy descriptions

However, for personal health records (PHRs) in cloud computing environments, the PHR service providers may not be in the same trust domains with the patients. Thus patient-centric privacy is hard to guarantee when full trust is placed on the cloud servers, since the patients lose physical control to their sensitive data. Therefore, the PHR needs to be encrypted in a way that enforces each patient's personalized privacy policy.

Patient Data Encryption: Efficient searching mechanism should satisfy the following properties: Search ability, which means the health server correctly returns the records which match the query, and Privacy, which means the patient can perform the search without revealing any information to the server (in this way, security is still guaranteed even if server has been compromised). Thus, we require that the server learn nothing about what query is being made or about the documents or keywords in the record. The server should only learn which encrypted documents must be returned.

Public Key Encryption: In a public key scheme, anyone can encrypt data without any secret information. Thus, in a public key PCE system, we can allow anyone to encrypt documents for the patient's file, and upload rights do not imply the ability to read other files in the same category. In practical terms, this means that doctors, devices and some other will be able to upload to a patient's record without receiving any secret key. However public key schemes tend to be slower, and when we also

require search ability or hidden labels, they seem to have inherent privacy weaknesses.

Privacy loss in search ability: There is also a significant loss in privacy in the public key option: since anyone can encrypt to any search term, any party with access to the patient's public key and history of query trapdoors can use these trapdoors to determine what keyword was being searched -for: he simply encrypts each possible keyword (using the patient's public key), and tests to see which one returns a match.

Symmetric Key Encryption: In a symmetric key encryption scheme, one must know the decryption key in order to encrypt data. Thus, in a symmetric key encryption system, anyone who can encrypt for a given category can also decrypt any files in that category. In practical terms, this means that the patient will have to issue an appropriate decryption key to a doctor or a device for a given category before they will be able to upload to this category. Furthermore, the doctor or device will also be able to read any data in that category. On the other hand, these schemes tend to be much more efficient and have stronger privacy guarantees.

III. Privacy Preserving and Security management PHR

In this section, we define the system model and then describe the implementations of the proposed privacy preserving and secure PHR scheme.

3.1 Current PHR System Model:

A) Trusted Authority (TA): It generates the public and secret key parameters for the privacy preserving policy. The trusted authority is responsible for attributes keys issuing, revoking, and updating. It grants differential access rights to individual users based on their attributes and roles. Trusted authority also maintains an index-table, where it stores the location of distributed data storage server. Authorized health service providers (e.g., Hospital, urgent care) are denoted as trusted parties.

B) Cloud service provider: It provides data outsourcing services and consists of data servers and data service manager. The main responsibility of the data storage server is to serve and retrieve data according to authorized users' request. Data service manager negotiates with health care service provider to control the access from outside users to the stored encrypted data.

C) Registered user: Patient who is registered to the trusted authority is considered as registered user. A registered user is responsible for defining attribute-based access policy and encrypting the sensitive PHR under the predefined policy before storing at the cloud-storage.

D) Data-access requester: Cloud users who request to access some specific PHR are called the data-access requester. The privacy preserving scheme ensures that any data-access requester can only decrypts the encrypted data if and only if he can successfully complete the access-policy.

3.2 Privacy Preserving System Overview:

We propose a variant of a privacy preserving CP-ABE scheme where the patient can encrypt her health records according to an access policy which has attributes issued by two trusted authorities: the

trusted authority (TA1) of the professional domain (PD) and the trusted authority (TA2) of the social domain (SD). The patient himself could also take the role of TA2. TA1 will authenticate users of the professional domain, and issue secret keys based on their attributes, while the patient might use the reputation of the users of the social domain to generate appropriate secret keys. For example, using our solution the patient can encrypt her health data such that a user who has the attribute General Practitioner issued from the TA1 of the professional domain, or the attribute friend issued by the patient can decrypt the encrypted data. Our scheme is suitable for the healthcare setting and has the following benefits:

• Allows a patient to store her PHRs in a protected form on an un-trusted commercial PHR server such that the access control policy is fully enforced. The patient encrypts the health data according to her access policy such that only the users who satisfy the access policy can decrypt the protected data.

• Helps the patient to share securely their PHRs with users from different security domains. This is because the access policy under which the data is encrypted can contain attributes issued from different trusted authorities.

• Removes the need for the patient to know the identity of the data recipient. The patient specifies only the attributes the recipient needs to have in order to access patient's data.

In the next section we demonstrate how to apply the proposed scheme to securely manage Personal Health Records (PHRs). The central issue here is how to achieve strong privacy guarantee for the owners. Consider a straightforward application of the CP-ABE scheme, where each AA in a PUD corresponds to an organization such as a health care provider, who defines all the attributes of its staffs and runs an independent ABE system. It is simple for an owner to realize complex access policies. If she wants to allow physicians from multiple hospitals to view one of her PHR file, she can include multiple sets of ciphertext components; each set encrypted using one of the hospital's ABE public keys. However, if any of the authorities (hospitals) misbehave, it can decrypt all the data of owners who allow access to users in that hospital. This is clearly against the patient-centric privacy concept. In addition, this method is not efficient since the policies for the three hospitals are duplicated, which makes the ciphertext long.

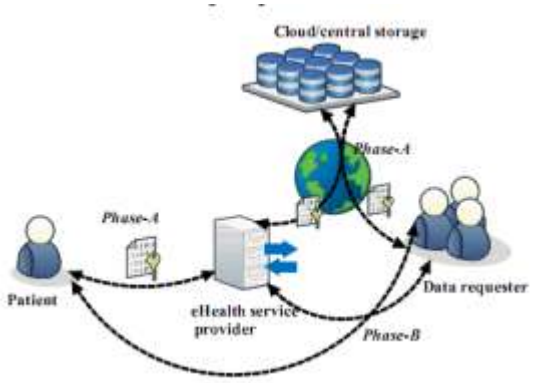

Figure 2. Data Access Policy flow architecture

To solve the above problems, we adopt the multi-authority ABE (MA-ABE) proposed by Chase et.al. [4], where each authority governs a disjoint set of attributes in distributive manner. An independent MA-ABE system is ran for each PUD, where there are multiple AAs in each of them; while each PSD (owner) runs the KP-ABE proposed. In each PUD,

there is no longer a central authority (CA) and any coalition of up to corrupted N −2 AAs cannot break the security of the system thanks to MA-ABE.

However, in MA-ABE the access policies are enforced in users' secret keys, and the policies are fixed once the keys are distributed which is not convenient for owners to specify their own policies. By our design, we show that by agreeing upon the formats of the key-policies and specifying which attributes are required in the ciphertext, the supported policy expressions enjoy some degree of flexibility from the encryptions point of view.

IV. Experimental Analysis

In this section, we analyze the security of proposed access control mechanisms. First, the GPSW and MA-ABE schemes are proven to be secure in [12] and respectively. Especially, the encrypted data is confidential to non-authorized users. Also, they are both resistant to user collusion, and MA-ABE is further resistant to collusion among up to N-2 AAs in one PUD. This implies that strong privacy guarantee is achieved through file encryption. Second, for the write access enforcement, the one-way property of the hash chain ensures that a writer can only obtain write keys for the time period that he is authorized for.

Performance Analysis: The performance analysis is summarized in Table. 3. We compare our solution with that of [22] which uses CP-ABE, and a single public authority is used. m is the number of PUDs, while Ni is the number of PAAs in the i^{th} PUD. Note that, the key management complexity is in terms of the number of interactions during key distribution. For ciphertext length comparison, for our scheme the access policy for each PUD is restricted to conjunctive form: Ppub := P1 ∧ ... ∧Pm, where each Pi is a boolean clause consisting of "∧"and"∨". The number of ciphertext components related to the PUDs.

Apart from those, for each owner, to change access policies and enable emergency access, 2 additional group elements (s and d) shall be locally stored for each encrypted PHR file, which is quite small. The result for [22]'s scheme is derived based on the same access policy to that in our scheme; it is a lower bound due to the lack of wildcard. Finally, the computational overhead in our scheme is low, since the decryption operation can be mostly delegated to the server.

V. Conclusion

In this paper, we have proposed a novel framework of access control to realize patient-centric privacy for personal health records in cloud computing. Considering partially trustworthy cloud servers, we argue that patients shall have full control of their own privacy through encrypting their PHR files to allow fine-grained access. The framework addresses the unique challenges brought by multiple PHR owners and users, in that we greatly reduce the complexity of key management when the number of owners and users in the system is large. We utilize multi-authority attribute-based encryption to encrypt the PHR data, so that patients can allow access not only by personal users, but also various users from different public domains with different professional roles, qualifications and affiliations.

VI. References

1) S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in IEEE INFOCOM'10, 2010.
2) C. Dong, G. Russello, and N. Dulay, "Shared and searchable encrypted data for untrusted servers," in Journal of Computer Security, 2010.
3) V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in CCS '06, 2006, pp. 89–98.
4) M. Li, W. Lou, and K. Ren, "Data security and privacy in wireless body area networks," IEEE Wireless Communications Magazine, Feb. 2010.
5) H. Harney, A. Colgrove, and P. D. McDaniel, "Principles of policy in secure groups," in Proc. of NDSS'01, 2001.
6) P. D. McDaniel and A. Prakash, "Methods and limitations of security policy reconciliation," in Proc. of SP'02, 2002.
7) T. Yu and M. Winslett, "A unified scheme for resource protection in automated trust negotiation," in Proc. of SP'03, 2003.
8) Benaloh, J., Chase, M., Horvitz, E., Lauter, K.: Patient controlled encryption: ensuring privacy of electronic medical records. In: CCSW 2009: Proceedings of the 2009 ACM workshop on Cloud computing security, pp. 103–114 (2009).
9) Mandl, K.D., Szolovits, P., Kohane, I.S.: Public standards and patients' control: how to keep electronic medical records accessible but private. BMJ 322(7281), 283 (2001).
10) Wang,W., Li, Z., Owens, R., Bhargava, B.: Secure and efficient access to outsourced data. In: CCSW 2009, pp. 55–66 (2009).
11) M. Barua, M. S. Alam, X. Liang, and X. Shen, "Secure and quality of service assurance scheduling scheme for wban with application to ehealth," in Wireless Communications and Networking Conference (WCNC), 2011 IEEE, Cancun, Quintana-Roo, Mexico, 2011, pp. 1 –5.
12) R. Lu, X. Lin, X. Liang, and X. Shen, "A secure handshake scheme with symptoms-matching for mhealthcare social network," Mobile Networks and Applications, pp. 1–12, 2010.