International Journal of Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.03,Issue.38
November-2014,
Pages:7643-7647

www.ijsetr.com

Utilization of Personal Health Records in Secured Manner using Attribute Based Encryption

V. DURGA RAMBABU[1], R. CHANDRA SEKHAR[2]

[1]PG Scholar, Dept of CSE, Kakinada Institute of Engineering &Technology, JNTUK, AP, India,
Email: rambabu.durga@gmail.com.

[2]Asst Prof, Dept of CSE, Kakinada Institute of Engineering &Technology, JNTUK, AP, India,
Email: sekharrayudu1221@gmail.com.

Abstract: The cloud computing paradigm has achieved widespread adoption in recent years. Its success is due largely to customers' ability to use services on demand with a pay-as-you go pricing model, which has proved convenient in many respects. Low costs and high flexibility make migrating to the cloud compelling. Several schemes employing attribute-based encryption (ABE) have been proposed for access control of outsourced data in cloud computing; In this paper, we propose hierarchical attribute-set-based encryption (HASBE) by extending cipher text-policy attribute-set-based encryption (ASBE) with a hierarchical structure of users. The proposed scheme not only achieves scalability due to its hierarchical structure, but also inherits flexibility and fine-grained access control in supporting compound attributes of ASBE. In addition, HASBE employs multiple value assignments for access expiration time to deal with user revocation more efficiently than existing schemes. We formally prove the security of HASBE based on security of the cipher text-policy attribute-based encryption (CP-ABE) scheme by Bettencourt and analyze its performance and computational complexity. We implement our scheme and show that it is both efficient and flexible in dealing with access control for outsourced data in cloud computing with comprehensive experiments.

Keywords: Personal Health Records, Cloud Computing, Data Privacy, fine-Grained Access Control, Attribute-Based Encryption.

I. INTRODUCTION

Cloud computing provides a whole new way to store, access, and exchange information, and introduces new service patterns and business opportunities for network providers and enterprises. Due to the expensive intercommunication among data centers, the difficulty of managing diverse backups, and the cost of operating several data centers, cloud resources are mostly limited to centralized architectures. However, achieving low latency and guaranteeing end-to-end performance is two main challenges in todays cloud computing. Meanwhile, PON-based broadband access networks have been widely considered as one of the most promising solutions for the future telecom access platform that supports residential, enterprise, and mobile backhaul services. The latest research has achieved terabit symmetric capacity that can support up to 800 ONUs at gigabit rates. Therefore, deploying cloud services based on PON access networks provides flexibility in moving services closer to customers so as to ensure low latency and guarantee quality of experience for customers. In this article, we propose, by incorporating PON OLT and ONU physical resources, an infrastructure as a service architecture that enables the delivery of cloud services to end users over PON access networks Which could impede its wide adoption. The main concern is about whether the patients could actually control the sharing of their sensitive personal health information (PHI), especially when they are stored on a third-party server which people may not fully trust. On the one hand, although there exist healthcare regulations such as HIPAA which is recently amended to incorporate business associates [4], cloud providers are usually not covered entities [5].

On the other hand, due to the high value of the sensitive personal health information (PHI), the third-party storage servers are often the targets of various malicious behaviors which may lead to expo- sure of the PHI. As a famous incident, a Department of Veterans Affairs database containing sensitive PHI of 26.5 million military veterans, including their social security numbers and health problems was stolen by an employee who took the data home without authorization [6]. To ensure patient-centric privacy control over their own PHRs, it is essential to have fine-grained data access control mechanisms that work with semi-trusted servers. A feasible and promising approach would be to encrypt the data before outsourcing. Basically, the PHR owner herself should decide how to encrypt her files and to allow which set of users to obtain access to each file. A PHR file should only be available to the users who are given the

corresponding decryption key, while remain confidential to the rest of users. Furthermore, the patient shall always retain the right to not only grant, but also revoke access privileges when they feel it is necessary [7]. However, the goal of patient-centric privacy is often in conflict with scalability in a PHR system. The authorized users may either need to access the PHR for personal use or professional purposes. Examples of the former are family member and friends, while the latter can be medical doctors, pharmacists, and researchers, etc. We refer to the two categories of users as personal and professional users, respectively.

The latter has potentially large scale; should each owner herself be directly responsible for managing all the professional users, she will easily be overwhelmed by the key management overhead. In addition, since those users' access requests are generally unpredictable, it is difficult for an owner to determine a list of them. On the other hand, different from the single data owner scenario considered in most of the existing works [8], [9], in a PHR system, there are multiple owners who may encrypt according to their own ways, possibly using different sets of cryptographic keys. Letting each user obtain keys from every owner whose PHR she wants to read would limit the accessibility since patients are not always online. An alternative is to employ a central authority (CA) to do the key management on behalf of all PHR owners, but this requires too much trust on a single authority (i.e., cause the key escrow problem). In this paper, we endeavor to study the patient- centric, secure sharing of PHRs stored on semi-trusted servers, and focus on addressing the complicated and challenging key management issues. In order to protect the personal health data stored on a semi-trusted server, we adopt attribute-based encryption (ABE) as the main encryption primitive.

Using ABE, access policies are expressed based on the attributes of users or data, which enables a patient to selectively share her PHR among a set of users by encrypting the file under a set of attributes, without the need to know a complete list of users. The complexities per encryption, key generation and decryption are only linear with the number of attributes involved. However, to integrate ABE into a large-scale PHR system, important issues such as key management scalability, dynamic policy updates, and efficient on-demand revocation are non-trivial to solve, and remain largely open up-to-date. To this end, we make the following main contributions:

- We propose a novel ABE-based framework for patient-centric secure sharing of PHRs in cloud computing environments, under the multi-owner settings. To address the key management challenges, we conceptually divide the users in the system into two types of domains, namely public and personal domains. In particular, the majority professional users are managed distributively by attribute authorities in the former, while each owner only needs to manage the keys of a small number of users in her personal domain. In this way, our framework can simultaneously handle different types of PHR sharing applications' requirements, while incurring minimal key management overhead for both owners and users in the system. In addition, the framework enforces write access control, handles dynamic policy updates, and provides break-glass access to PHRs under emergence scenarios.
- In the public domain, we use multi-authority ABE (MA-ABE) to improve the security and avoid key escrow problem. Each attribute authority (AA) in it governs a disjoint subset of user role attributes, while none of them alone is able to control the security of the whole system. We propose mechanisms for key distribution and encryption so that PHR owners can specify personalized fine-grained role-based access policies during file encryption. In the personal domain, owners directly assign access privileges for personal users and encrypt a PHR file under its data attributes. Furthermore, we enhance MA-ABE by putting forward an efficient and on-demand user/attribute revocation scheme, and prove its security under standard security assumptions. In this way, patients have full privacy control over their PHRs.
- We provide a thorough analysis of the complexity and scalability of our proposed secure PHR sharing solution, in terms of multiple metrics in computation, communication, storage and key management.

Compared with the preliminary version of this paper [1], there are several main additional contributions: (1) We clarify and extend our usage of MA-ABE in the public domain, and formally show how and which types of user-defined file access policies are realized. (2) We clarify the proposed revocable MA-ABE scheme, and provide a formal security proof for it. (3) We carry out both real-world experiments and simulations to evaluate the performance of the proposed solution in this paper.

II. RELATED WORK

This paper is mostly related to works in cryptographically enforced access control for out sourced data and attribute based encryption. To realize fine-grained access control, the traditional public key encryption (PKE) based schemes [8], [10] either incur high key management overhead, or require encrypting multiple copies of a file using different users' keys. To improve upon the scalability of the above solutions, one-to-many encryption methods such as ABE can be used. In Goyal et. al's seminal paper on ABE [11], data is encrypted under a set of attributes so that multiple users who possess proper keys can decrypt. This potentially makes encryption and key management more efficient [12]. A fundamental property of ABE is preventing against user collusion. In addition, the encryptor is not required to know the ACL.

A. ABE for Fine-grained Data Access Control

A number of works used ABE to realize fine-grained access control for outsourced data [13], [14], [9], [15].Especially,

there has been an increasing interest in applying ABE to secure electronic healthcare records (EHRs). Recently, Narayan et al. proposed an attribute-based infrastructure for EHR systems, where each patient's EHR files are encrypted using a broadcast variant of CP-ABE [16] that allows direct revocation. However, the cipher text length grows linearly with the number of unrevoked users. In [17], a variant of ABE that allows delegation of access rights is proposed for encrypted EHRs. Ibrahim et.al. [18] Applied cipher text policy ABE (CP-ABE) [19] to manage the sharing of PHRs, and introduced the concept of social/professional domains. In [20], Akinyele et al. investigated using ABE to generate self-protecting EMRs, which can either be stored on cloud servers or cell phones so that EMR could be accessed when the health provider is offline. However, there are several common drawbacks of the above works. First, they usually assume the use of a single trusted authority (TA) in the system. This not only may create a load bottleneck, but also suffers from the key escrow problem since the TA can access all the encrypted files, opening the door for potential privacy exposure.

In addition, it is not practical to delegate all attribute management tasks to one TA including certifying all users' attributes or roles and generating secret keys. In fact, different organizations usually form their own (sub) domains and become suitable authorities to define and certify different sets of attributes belonging to their (sub) domains (i.e., divide and rule). For example, a professional association would be responsible for certifying medical specialties, while a regional health provider would certify the job ranks of its staffs. Second, there still lacks an efficient and on-demand user revocation mechanism for ABE with the support for dynamic policy updates/changes, which are essential parts of secure PHR sharing. Finally, most of the existing works do not differentiate between the personal and public do- mains, which have different attribute definitions, key management requirements and scalability issues. Our idea of conceptually dividing the system into two types of domains is similar with that in [18], however a key difference is in [18] a single TA is still assumed to govern the whole professional domain.

Recently, Yu et al. (YWRL) applied key-policy ABE to secure outsourced data in the cloud [9], [15], where a single data owner can encrypt her data and share with multiple authorized users, by distributing keys to them that contain attribute-based access privileges. They also propose a method for the data owner to revoke a user efficiently by delegating the updates of affected cipher texts and user secret keys to the cloud server. Since the key update operations can be aggregated over time, their scheme achieves low amortized overhead. However, in the YWRL scheme, the data owner is also a TA at the same time. It would be inefficient to be applied to a PHR system with multiple data owners and users, because then each user would receive many keys from multiple owners, even if the keys contain the same sets of attributes. On the other hand, Chase and Chow [21] proposed a multiple-authority ABE (CC MA- ABE)solution in which multiple TAs, each governing a different subset of the system's users' attributes, generate user secret keys collectively. A user needs to obtain one part of her key from each TA. This scheme prevents against collusion among at most $N - 2$ TAs, in addition to user collusion resistance. However, it is not clear how to realize efficient user revocation. In addition, since CC MA-ABE embeds the access policy in users' keys rather than the ciphertext, a direct application of it to a PHR system is non-intuitive, as it is not clear how to allow data owners to specify their file access policies. We give detailed overviews to the YWRL scheme and CCMA- ABE scheme in the supplementary material.

B. Revocable ABE

It is a well-known challenging problem to revoke users/ attributes efficiently and on-demand in ABE. Traditionally this is often done by the authority broadcasting periodic key updates to unrevoked users frequently [13], [22], which does not achieve complete backward/forward security and is less efficient. Recently, [23] and [24] proposed two CP-ABE schemes with immediate attribute revocation capability, instead of periodical revocation. However, they were not designed for MA- ABE. In addition, Ruj et al. [25] proposed an alternative solution for the same problem in our paper using Lewko and Waters's (LW) decentralized ABE scheme [26]. The main advantage of their solution is, each user can obtain secret keys from any subset of the TAs in the system, in contrast to the CC MA-ABE. The LW ABE scheme enjoys better policy expressiveness, and it is extended by [25] to support user revocation. On the downside, the communication overhead of key revocation is still high, as it requires a data owner to transmit an updated cipher text component to every non-revoked user. They also do not differentiate personal and public domains.

In this paper, we bridge the above gaps by proposing a unified security framework for patient-centric sharing of PHRs in a multi-domain, multi-authority PHR system with many users. The framework captures application- level requirements of both public and personal use of a patient's PHRs, and distributes users' trust to multiple authorities that better reflects reality. We also propose a suite of access control mechanisms by uniquely combining the technical strengths of both CC MA-ABE [21] and the YWRL ABE scheme [9]. Using our scheme, patients can choose and enforce their own access policy for each PHR file, and can revoke a user without involving high overhead. We also implement part of our solution in a prototype PHR system.

III.FRAMEWORK FOR PATIENT-CENTRIC, SECURE AND SCALABLE PHR SHARING

In this section, we describe our novel patient-centric secure data sharing framework for cloud-based PHR systems.

A. Problem Definition

We consider a PHR system where there are multiple PHR owners and PHR users. The owners refer to patients who have full control over their own PHR data, i.e., they can create, manage and delete it. There is a central server belonging to the PHR service provider that stores all the owners' PHRs. The users may come from various aspects; for example, a friend, a caregiver or a researcher. Users access the PHR documents through the server in order to read or write to someone's PHR, and a user can simultaneously have access to multiple owners' data. A typical PHR system uses standard data formats. For example, continuity-of-care (CCR)(based on XML data structure), which is widely used in representative PHR systems including Indivo [27], an open-source PHR system adopted by Boston Children's Hospital. Due to the nature of XML, the PHR files are logically organized by their categories in a hierarchical way [8], [20].

Security Model: In this paper, we consider the server to be semi-trusted, i.e., honest but curious as those in [28] and [15]. That means the server will try to find out as much secret information in the stored PHR files as possible, but they will honestly follow the protocol in general. On the other hand, some users will also try to access the files beyond their privileges. For example, a pharmacy may want to obtain the prescriptions of patients for marketing and boosting its profits. To do so, they may collude with other users, or even with the server. In addition, we assume each party in our system is preloaded with a public/private key pair, and entity authentication can be done by traditional challenge-response protocols.

B. Requirements

To achieve "patient-centric" PHR sharing, a core requirement is that each patient can control who are authorized to access to her own PHR documents. Especially, user-controlled read/write access and revocation are the two core security objectives for any electronic health record system, pointed out by Mandl et. al. [7] in as early as 2001. The security and performance requirements are summarized as follows

Data Confidentiality: Unauthorized users (including the server) who do not possess enough attributes satisfying the access policy or do not have proper key access privileges should be prevented from decrypting a PHR document, even under user collusion. Fine-grained access control should be enforced, meaning different users are authorized to read different sets of documents.

On-Demand Revocation: Whenever a user's attribute is no longer valid, the user should not be able to access future PHR files using that attribute. This is usually called attribute revocation, and the corresponding security property is forward secrecy [23]. There is also user revocation, where all of a user's access privileges are revoked.

Write Access Control: We shall prevent the unauthorized contributors to gain write-access to owners' PHRs, while the legitimate contributors should access the server with accountability. The data access policies should be flexible, i.e., dynamic changes to the pre-defined policies shall be allowed, especially the PHRs should be accessible under emergency scenarios.

Scalability, Efficiency and Usability: The PHR system should support users from both the personal do- main and public domains. Since the set of users from the public domain may be large in size and unpredictable, the system should be highly scalable, in terms of complexity in key management, communication, computation and storage. Additionally, the owners' efforts in managing users and keys should be minimized to enjoy usability.

C. Details of the Proposed Fram work

In our framework, there are multiple SDs, multiple owners, multiple AAs, and multiple users. In addition, two ABE systems are involved: for each PSD the YWRL's revocable KP-ABE scheme [9] is adopted; for each PUD, our proposed revocable MA-ABE scheme (described in Sec. 4) is used. The framework is illustrated in Fig.1. We term the users having read and write access as data readers and contributors, respectively.

PHR Encryption and Access: The owners upload ABE-encrypted PHR files to the server ((3)). Each owner's PHR file is encrypted both under a certain fine-grained and role-based access policy for users from the PUD to access, and under a selected set of data attributes that allows access from users in the PSD. Only authorized users can decrypt the PHR files, excluding the server. For improving efficiency, the data attributes will include all the intermediate file types from a leaf node to the root.

User Revocation: Here we consider revocation of a data reader or her attributes/access privileges. There are several possible cases: 1) revocation of one or more role attributes of a public domain user; 2) revocation of a public domain user which is equivalent to revoking all of that user 's attributes. These operations are done by the AA that the user belongs to, where the actual computations can be delegated to the server to improve efficiency ((8)). 3) Revocation of a personal domain user's access privileges; 4) revocation of a personal domain user. These can be initiated through the PHR owner's client application in a similar way.

Break-Glass: When an emergency happens, the regular access policies may no longer be applicable. To andle this situation, break-glass access is needed to access the victim's PHR. In our framework, each owner's PHR's access right is also delegated to an emergency department (ED, (6)). To prevent from abuse of break-glass option, the emergency staff needs to contact the ED to verify her identity and the emergency situation, and obtain temporary read keys ((7)). After the emergency is over, the patient can revoke the emergent access via the ED.

IV. CONCLUSION

In this paper, we have proposed a novel framework of secure sharing of personal health records in cloud computing. This paper aims at fine-grained data access control in cloud computing. One challenge in this context is to achieve fine grainedness, data confidentiality, and scalability simultaneously, which is not provided by current work. In this paper we propose a scheme to achieve this goal by exploiting KPABE and uniquely combining it with techniques of proxy re-encryption and lazy re-encryption. Moreover, our proposed scheme can enable the data owner to delegate most of computation overhead to powerful cloud servers. Confidentiality of user access privilege and user secret key accountability can be achieved. Formal security proofs show that our proposed scheme is secure under standard cryptographic models. We utilize ABE to encrypt the PHR data, so that patients can allow access not only by personal users, but also various users from public domains with different professional roles, qualifications and affiliations. Furthermore, we enhance an existing MA-ABE scheme to handle efficient and on-demand user revocation, and prove its security. Through implementation and simulation, we show that our solution is both scalable and efficient.

V. FUTURE ENHANCEMENT

Taking into consideration moderately responsible cloud servers, we dispute that to completely apprehend the patient-centric model, patients shall have extensive manage of their own privacy through enciphering their Personal Health Record files to permit fine-grained access. The method addresses the distinctive goals brought by various Personal Health Record users and owners, in that we completely decrease the complication of key management while enhance the privacy assurance compared with prior works. We use Attribute Based Encryption to encipher the Personal Health Record data, hence that patients can permit access not only by personal users, but also many users from public domains with different professional roles, affiliations and qualifications. In addition, we enhance an existing Multi Authority Attribute Based Encryption scheme to manage on-demand user revocation, efficient, and prove its security.

VI. REFERENCES

[1]M. Li, S. Yu, K. Ren, and W. Lou, "Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings," in SecureComm'10, Sept. 2010, pp. 89–106.

[2] H. Lo¨ hr, 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 comput- ing," in ICDCS '11, Jun. 2011.

[4] "The health insurance portability and accountability act." [Online] Available: http://www.cms.hhs.gov/ HIPAAGenInfo/01 Overview.asp.

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

[6] "At risk of exposure – in the push for electronic medical records, concern is growing about how well privacy can be safeguarded," 2006. [Online]. Available: http://articles.latimes.com /2006/jun/26/health/ he-privacy26

[7] K. D. Mandl, P. Szolovits, and I. S. Kohane, "Public standards and patients' control: how to keep electronic medical records accessible but private," BMJ, vol. 322, no. 7281, p. 283, Feb. 2001.

[8] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient controlled encryption: ensuring privacy of electronic medical record- s," in CCSW '09, 2009, pp. 103–114.

Author's Profile:

V.DURGA RAMBABU, PG Scholar **(**Department of Computer Science & Engineering, Kakinada Institute of Engineering &Technology (KIET), JNTUKm, Korangi, currently he is pursuing his M.Tech (CSE) (12B21D25806) from this college he received his graduation from sri vasavi engineering college tadepalligudem. In the year 2012. His areas of interest are cloud computing. Email: rambabu.durga@gmail.com**).**

R. CHANDRASEKHAR, Is working as Assistant Professor in KIET.He has five years experience. He completed his b.tech from KIEt in 2007. He completed his M.tech from GIET rajamandry in 2010. He had published his paper international journal of computer science & technology.
Email: sekharrayudu1221@gmail.com)