Attribute-Based Encryption for Securely Sharing of Personal Health Records in Cloud Computing

Shaik Shahina[1], Guntapalli Minni [2]

[1] M.Tech(CSE), Nimra College of Engineering and Technology, A.P., India.

[2]Asst. Professor, Dept. of Computer Science & Engineering, Nimra College of Engineering and Technology, A.P., India.

Abstract— Outsourcing Medical Details is a common phenomenon. Personal health record (PHR) is an emerging patient-centric model of health information exchange, which is often outsourced to be stored at a third party, such as cloud providers. However, there have been wide privacy concerns as personal health information could be exposed to those third party servers and to unauthorized parties. To assure the patients' control over access to their own PHRs, it is a promising method to encrypt the PHRs before outsourcing. Yet, issues such as risks of privacy exposure, scalability in key management, flexible access, and efficient user revocation, have remained the most important challenges toward achieving fine-grained, cryptographically enforced data access control. In this paper, we propose a novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semitrusted servers. To achieve fine-grained and scalable data access control for PHRs, we leverage attribute-based encryption (ABE) techniques to encrypt each patient's PHR file. Different from previous works in secure data outsourcing, we focus on the multiple data owner scenario, and divide the users in the PHR system into multiple security domains that greatly reduces the key management complexity for owners and users. A high degree of patient privacy is guaranteed simultaneously by exploiting multiauthority ABE. Our scheme also enables dynamic modification of access policies or file attributes, supports efficient on-demand user/attribute revocation and break-glass access under emergency scenarios. Extensive analytical and experimental results are presented which show the security, scalability, and efficiency of our proposed scheme.

Keywords— Cloud computing, Personal health records, data privacy, fine-grained access control, attribute-based encryption.

I. Introduction

Now a days, personal health record (PHR) has been emerging as a patient-centric model of health information exchange. A PHR service allows a patient to create, manage, and control her personal health data in one place through the web, which has made the storage, retrieval, and sharing of the medical information more efficient. Especially, each patient is promised the full control of her medical records and can share her health data with a wide range of users, including healthcare providers, family members or friends. Due to the high cost of building and maintaining specialized data centers, many PHR services are outsourced to or provided by third-party service providers, for example, Microsoft HealthVault[1] Recently, architectures of storing PHRs in cloud computing have been proposed in [2]. While it is exciting to have convenient PHR services for everyone, there are many security and privacy risks 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, cloud providers are usually not covered entities. On the other hand, due to the high value of the sensitive PHI, the third-party storage servers are often the targets of various malicious behaviors which may lead to exposure 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 [3]. To ensure patient-centric privacy control over their own PHRs, it is essential to have fine-grained data access control mechanisms that work with semitrusted servers.

In this paper, we study the patient-centric, secure sharing of PHRs stored on semitrusted servers, and focus on addressing the complicated and challenging key management issues. In order to protect the personal health data stored on a semitrusted 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. 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 nontrivial to solve, and remain largely open up-to-date.

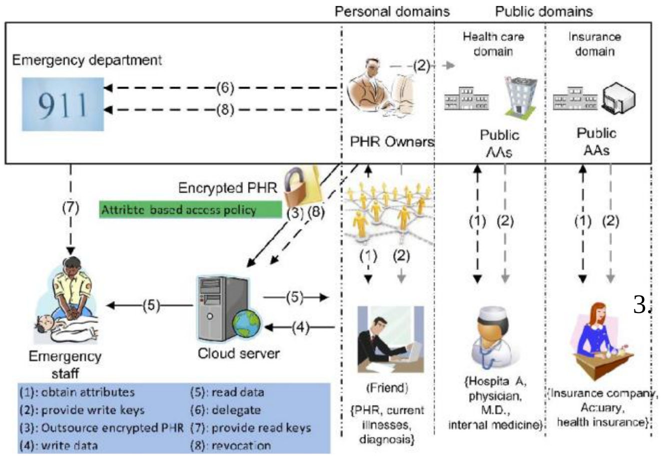


Figure 1: The proposed framework for patient-centric, secure and scalable PHR sharing on semitrusted storage under multiowner settings.

To this end, we make the following main contributions:

1. We propose a novel ABE-based framework for patient-centric secure sharing of PHRs in cloud computing environments, under the multiowner 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 (PSDs). 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.

2. In the public domain, we use multiauthority 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.

3. 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. We also compare our scheme to several previous ones in complexity, scalability and security. Furthermore, we demonstrate the efficiency of our scheme by implementing it on a modern workstation and performing experiments/simulations.

II. Related work

This paper is mostly related to works in cryptographically enforced access control for outsourced data and attribute based encryption. To realize fine-grained access control, the traditional public key encryption (PKE)-based schemes [4] 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, data are 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 [5]. A fundamental property of ABE is preventing against user collusion. In addition, the encryptor is not required to know the ACL.

A number of works used ABE to realize fine-grained access control for outsourced data 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 ciphertext 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. Ibraimi et al. [18] applied ciphertext policy ABE (CP-ABE) [6] to manage the sharing of PHRs, and introduced the concept of social/professional domains. In [7], Akinyele et al. investigated using ABE to generate self-protecting EMRs, which can either be stored on cloud servers or cellphones so that EMR could be accessed when the health provider is offline.

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, an open-source PHR system adopted by Boston Children's Hospital. Due to the nature of XML, thePHRfiles are logically organized by their categories in a hierarchical way [7].

III. Proposed work

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

Security Model

In this paper, we consider the server to be semitrusted, i.e., honest but curious as those in [8]. 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.

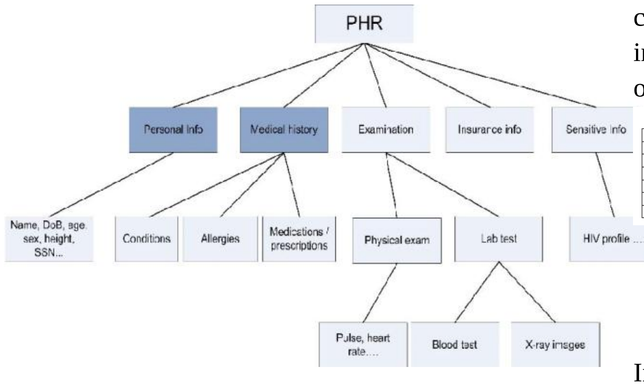
Requirements

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. 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 [9]. 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 predefined 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 domain 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.



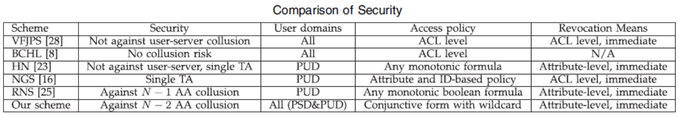
Enforce Write Access Control

If there is no restrictions on write access, anyone may write to someone's PHRusing only public keys, which is undesirable. By granting write access, we mean a data contributor should obtain proper authorization from the organization she is in (and/or from the targeting owner), which shall be able to be verified by the server who grants/rejects write access. A naive way is to let each contributor obtain a signature from her organization every time she intends to write. Yet this requires the organizations be always online. The observation is that, it is desirable and practical to authorize according to time periods whose granularity can be adjusted. For example, a doctor should be permitted to write only during her office hours; on the other hand, the doctor must not be able to write to patients that are not treated by her. Therefore, we combine signatures with the hash chain technique to achieve our goals.

Handle Dynamic Policy Changes

Our scheme should support the dynamic add/modify/ delete of part of the document access policies or data attributes by the owner. For example, if a patient does not want doctors to view her PHR after she finishes a visit to a hospital, she can simply delete the ciphertext components corresponding to attribute "doctor" in her PHR files. Adding and modification of attributes/access policies can be done by proxy reencryption techniques [10]; however, they are expensive. To make the computation more efficient, each owner could store the random number s used in encrypting the FEK3 of each document on her own computer, and construct new ciphertext components corresponding to added/changed attributes based on s. To reduce the storage cost, the owner can merely keep a random seed s0 and generate the s for each encrypted file from s0, such as using a pseudorandom generator. Thus, the main computational overhead to modify/add one attribute in the ciphertext is just one modular exponentiation operation.

Comparison of Security



Scheme	Security	User domains	Access policy	Revocation Means
VFJPS [28]	Not against user-server collusion	All	ACL level	ACL level, immediate
BCHL [8]	No collusion risk	All	ACL level	N/A
HN [23]	Not against user-server, single TA	PUD	Any monotonic formula	Attribute-level, immediate
NGS [16]	Single TA	PUD	Attribute and ID-based policy	ACL level, immediate
RNS [25]	Against $N-1$ AA collusion	PUD	Any monotonic boolean formula	Attribute-level, immediate
Our scheme	Against $N-2$ AA collusion	All (PSD&PUD)	Conjunctive form with wildcard	Attribute-level, immediate

IV. Conclusion

In this paper, we have proposed a novel framework of secure sharing of personal health records in cloud computing. Considering partially trustworthy cloud servers, we argue that to fully realize the patient-centric concept, patients shall have complete 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 while enhance the privacy guarantees compared with previous works. 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.

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," Proc. Sixth Int'l ICST Conf. Security and Privacy in Comm. Networks (SecureComm '10), pp. 89-106, Sept. 2010.

[2] H. Lo¨ hr, A.-R. Sadeghi, and M. Winandy, "Securing the E-Health Cloud," Proc. First ACM Int'l Health Informatics Symp. (IHI '10), pp. 220-229, 2010.

[3] "At Risk of Exposure - in the Push for Electronic Medical Records, Concern Is Growing About How Well Privacy Can Be Safeguarded," http://articles.latimes.com/2006/jun/26/health/he-privacy26, 2006.

[4] C. Dong, G. Russello, and N. Dulay, "Shared and Searchable Encrypted Data for Untrusted Servers," J. Computer Security, vol. 19, pp. 367-397, 2010.

[5] M. Li, W. Lou, and K. Ren, "Data Security and Privacy in Wireless Body Area Networks," IEEE Wireless Comm. Magazine, vol. 17, no. 1, pp. 51-58, Feb. 2010.

[6] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute- ased Encryption," Proc. IEEE Symp. Security and Privacy (SP '07), pp. 321-334, 2007.

[7] J.A. Akinyele, C.U. Lehmann, M.D. Green, M.W. Pagano, Z.N.J. Peterson, and A.D. Rubin, "Self-Protecting Electronic Medical Records Using Attribute-Based Encryption," Cryptology ePrint Archive, Report 2010/565, ttp://eprint.iacr.org/, 2010.

[8] S.D.C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-Encryption: Management of Access Control Evolution on Outsourced Data," Proc. 33rd Int'l Conf. Very Large Data Bases (VLDB '07), pp. 123- 34, 2007.

[9] J. Hur and D.K. Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems," IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 7, pp. 1214-1221, July 2011.

[10] X. Liang, R. Lu, X. Lin, and X.S. Shen, "Ciphertext Policy Attribute Based Encryption with Efficient Revocation," technical report, Univ. of Waterloo, 2010.