



## Secured Sharing Of Personal Health Records in Cloud Computing Using Attribute Based Encryption (ABE)

Dr. CS. Pillai.

Associate Professor & HOD,  
Dept of CSE, ACS College of Engineering,  
Bangalore, India

---

**Abstract:** *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 semi-trusted 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 multi-authority 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:** *Personal Health Record, Cloud Providers, Attribute Based Encryption*

---

### I. INTRODUCTION

In recent years, personal health record (PHR) has emerged 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<sup>1</sup>. Recently, architectures of storing PHRs in cloud computing have been proposed in.

While it is exciting to have convenient PHR services for everyone, there are many security and privacy risk 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 personal health information (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. 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 himself 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. 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 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 who's 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:

(1) 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.

(2) 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.

(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. METHODOLOGY

After careful analysis the system has been identified to have the following

- PHR OWNER
- Cloud Server
- Attribute based Access Policy
- Data confidentiality
- PHR OWNER

The main goal of our framework is to provide secure patient-centric PHR access and efficient key management at the same time. The key idea is to divide the system into multiple security domains (namely, public domains (PUDs) and personal domains (PSDs)) according to the different users' data access requirements. The PUDs consist of users who make access based on their professional roles, such as doctors, nurses and medical researchers. In practice, a PUD can be mapped to an independent sector in the society, such as the health care, government or insurance sector. For each PSD, its users are personally associated with a data owner (such as family members or close friends), and they make accesses to PHRs based on access rights assigned by the owner.

Each data owner (e.g., patient) is a trusted authority of her own PSD, who uses a KP-ABE system to manage the secret keys and access rights of users in her PSD. Since the users are personally known by the PHR owner, to realize patient-centric access, the owner is at the best position to grant user access privileges on a case-by-case basis. For PSD, data attributes are defined which refer to the intrinsic properties of the PHR data, such as the category of a PHR file. For the purpose of PSD access, each PHR file is labeled with its data attributes, while the key size is only linear with the number of file categories a user can access. Since the number of users in a PSD is often small, it reduces the burden for the owner. When encrypting the data for PSD, all that the owner needs to know is the intrinsic data properties.

### • Cloud Server

In this paper, we consider the server to be semi-trusted, i.e., honest but curious as those in and. 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.

- **Attribute based Access Policy**

In our framework, there are multiple SDs, multiple owners, multiple AAs, and multiple users. In addition, two ABE systems are involved. We term the users having read and write access as data readers and contributors, respectively.

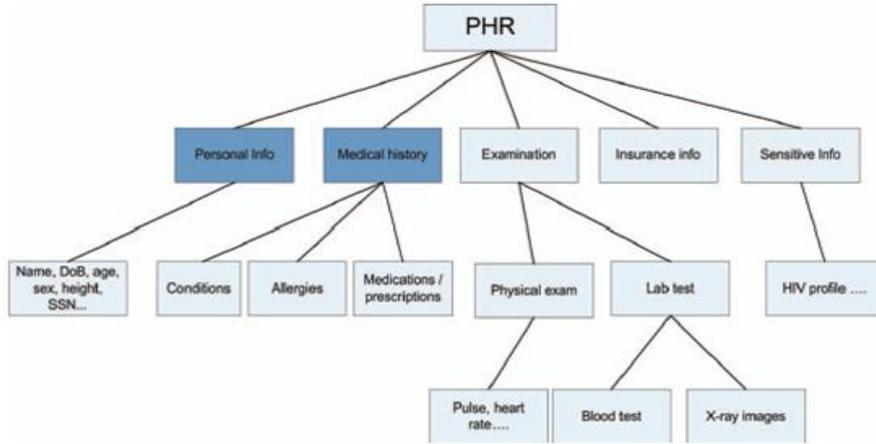


Fig 1.1 Attribute based Access Policy Module

- **Data confidentiality**

The owners upload ABE-encrypted PHR files to the server. 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.

### III. IMPLEMENTATION RESULTS



Fig1.2: PHR Login

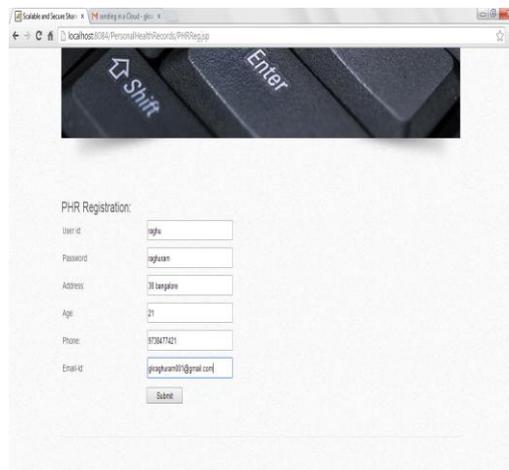


Fig1.3: PHR Registration

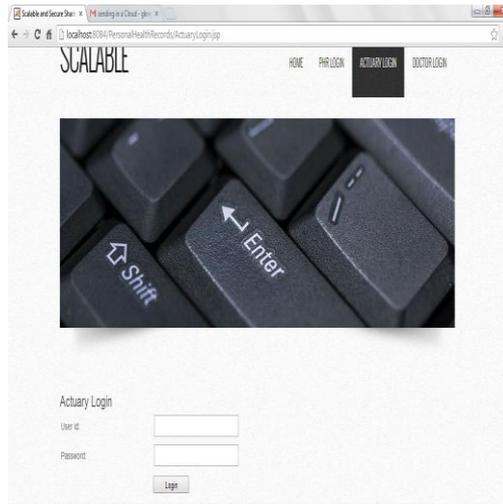


Fig1.4: Actuary Login



Fig1.5: Doctor Login

For your File1\_sudhanshijbsjdb.txtSecret key=0011000101011110110011011101  
 01011001000110100001100001011011100111001101010000110100101  
 10101001100010011100110101010011001000110001000101110011101  
 0001111000011101000011001000110001 MasterKey=107569 Public  
 Key=011100100110000101101101

Fig1.6:Key Mailed to User



Fig1.7: Doctor viewing Patient details

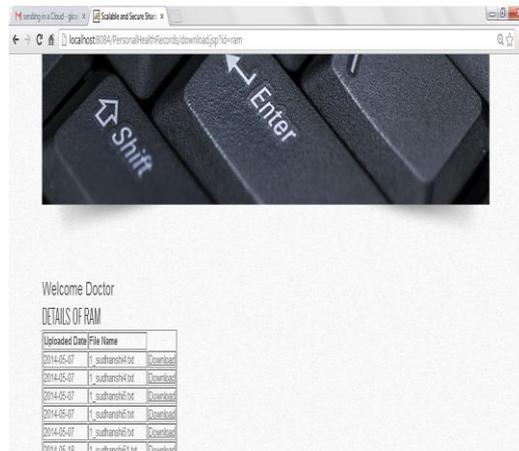


Fig1.8: Details of the Patient

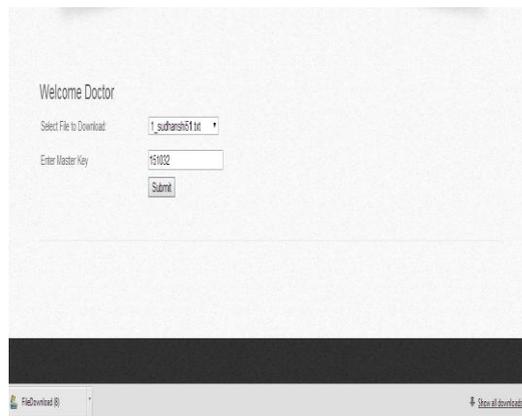


Fig1.9: Doctor downloading details of patient

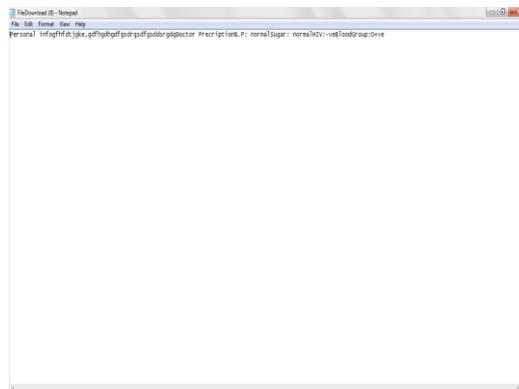


Fig1.10: Details of Patient by using Doctor's Master Key



Fig1.11: Emergency Login

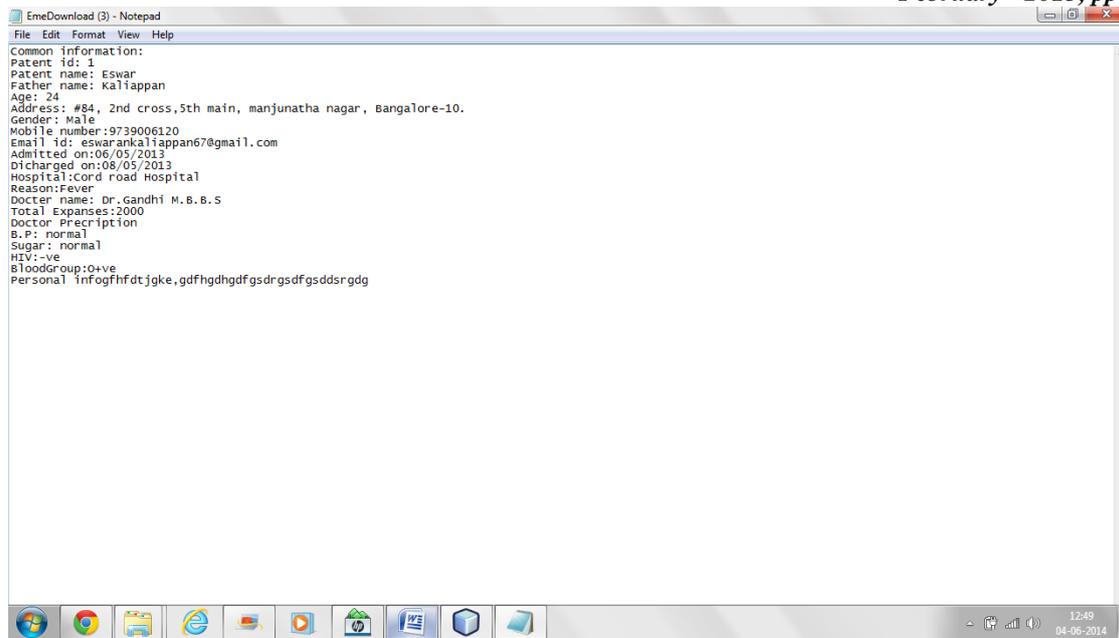


Fig1.12: Patient details viewed by Emergency

#### 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.

#### V. FUTURE ENHANCEMENT

In future 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. We can provide good security to our data's using encryption technique in cloud .In practice, the credentials from different organizations may be considered equally effective, in that case distributed ABE schemes, will be needed. We designate those issues as future works.

#### REFERENCES

- [1] Gnutella P2P Network. [www.gnutella.com](http://www.gnutella.com).
- [2] Kazaa P2P Network. [www.kazaa.com](http://www.kazaa.com).
- [3] TIB/Rendezvous. White paper, 1999.
- [4] S. Androutsellis-Theotokis and D. Spinellis. A Survey of Peer to-Peer Content Distribution Technologies. ACM Comput. Surv., 2004.
- [5] B. Arai, G. Das, D. Gunopulos, and V. Kalogeraki. Approximating Aggregation Queries in Peer-to-Peer Networks. In Proceedings of the 22nd International Conference on Data Engineering (ICDE), 2006.
- [6] R. Baldoni, C. Marchetti, A. Virgillito, and R. Vitenberg. Contentbased Publish-Subscribe over Structured Overlay Networks. In Proceedings of ICDCS , 2005.
- [7] G. Banavar, T. Chandra, B. Mukherjee, J. Nagarajarao, R. E. Strom, and D. C. Sturman. An Efficient Multicast Protocol for Content- Based Publish-Subscribe Systems. In Proceedings of ICDCS 1999, 1999.
- [8] T. Bu and D. F. Towsley. On Distinguishing between Internet Power Law Topology Generators. In INFOCOM, 2002.
- [9] A. Carzaniga, D. S. Rosenblum, and A. L. Wolf. Design and evaluation of a wide-area event notification service. ACM Transactions on Computer Systems, 19(3):332.383, 2001.
- [10] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker. Making Gnutella-like P2P Systems Scalable. In Proceedings of ACM SIGCOMM 2003, 2003.
- [11] J. Chen, L. Ramaswamy, and A. Meka. Message Diffusion in Unstructured Overlay Networks. In Proceedings of NCA, 2007.
- [12] P. Chirita, S. Idreos, M. Koubarakis, and W. Nejdl. Publish/ Subscribe for RDF-based P2P Networks. In Proceedings of the 1st European Semantic Web Symposium, May 2004.