Volume 4, No. 8, May-June 2013

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Towards Secure and Efficient Mobile Cloud Computing for Medical data sharing using Attribute based Encryption

Gargi. N, Sharada. K.A
Department of Computer Science and Engineering
East West Institute of Technology, Bangalore
gargi.n4@gmail.com, Sharadaa1234@gmail.com

Abstract: The healthcare sector represents one of the most important and growing industry in terms of support from IT. Existing healthcare systems are built on workflow that consists of paper medical records, duplicated test results, on digitized images, handwritten notes. Hospitals and providers are facing the risk of capacity shortage to securely store and share patient medical records and information. Multiple efforts are made to modernize medical records for greater efficiency, improved patient care, patient safety, and patient privacy and cost savings.. Health information Exchange (HIE) is the provision of exchanging healthcare information within or across organization. We implement attribute based encryption (ABE) to encrypt patients Personal Health record (PHR) file An Electronic Medical Record (EMR) is the effective capture, dissemination, and analysis of medical and health related information for a single patient and it will handle all the users query and give the online solutions via mobile cloud immediately.

Index Terms: Personal health records, cloud computing, attribute-based encryption

I. INTRODUCTION

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. We must encrypt the PHRs before outsourcing to ensure the security.

PHR service allows a patient to create, manage and control her personal health data in one place through the cloud server, which has made the storage, retrieval and sharing of the medical information more efficient. 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. The goal of patient-centric privacy is often in conflict with scalability in a PHR system. We propose a framework for Patient-centric secure sharing of PHRs in cloud computing environments, under the multi-owner settings [1]

To address the key management challenges, we conceptually divide the users in the system into two types of domains, namely public and personal domains. The authorized users may either need to access the PHR for personal use or professional purposes. 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 [2]

Health information Exchange (HIE) is used for exchanging healthcare information within or across organization. Eg: Interacting with lab or ordering tests/receive results, transmitting prescriptions from physicians to pharmacies, sharing patient health history between physicians, relaying data from patient's home medical devices to physicians and giving patients access to their health information.

Electronic Medical Records [3] are managed by individuals are known as Personal Health Records (PHRs) PHRs capture all personal health details, including diagnoses, X-Rays, and similar things into a single repository. Using EMRs, doctors can review patient histories and charts, obtain laboratory results, generate referrals for specialist consultations, prescribe medicines, and diagnose images all without the use of paper.

A. Existing system and Disadvantage:

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) [4], [5] based schemes 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. 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. A fundamental property of ABE is preventing against user collusion. In addition, the encryptor is not required.

The drawback of this system is that traditional encryption methods like AES, DES which had 32 bit were used where all the records were encrypted in one file which resulted in less security and efficiency.

B. Proposed system and its Advantages:

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) [6] 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 [4]

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..

Akinyele 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 [7] the advantage of the proposed system is that new encryption method such as RSA-128 bit is used for attribute based encryption for encryption and decryption. This improves the security to a greater extent. Each patient's personal health records are encrypted in different set of attributes and a secret key is generated for each PHR which ensures security, scalability and efficiency [8]

II. REQUIREMENT ANALYSIS

A. Functional Requirement:

Functional requirement defines a function of a software system or its component and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality that define what a system must accomplish.

Functional requirements of this system are:

a. The Doctor or User has to upload the patient Details to the Cloud server
b. The Cloud maintains the data Storage and has to authorize the valid remote users. If the Remote user is hacker then he has to block in the cloud server.
c. The Third party auditor has to maintain the Data Integrity and has to monitor the Cloud Server Activities.
d. The Remote user has to user Provisions and Secret Key. If anyone is wrong then he is detected as attacker.

B. Non-Functional Requirement:

Non – Functional requirements, as the name suggests, are those requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability, response time, performance. Many non-functional requirements relate to the system as whole rather than to individual system features.

Non-functional requirements of this system are:

a. Security: The system should allow a secured communication between data owners and remote user and cloud server.

b. Energy Efficiency: The energy consumed by the users to receive the information from the data owner should be efficient.

c. Reliability: The system should be reliable and must not degrade the performance of the existing system and should not lead to the hanging of the system.

III. HIGH LEVEL DESIGN

A. System Architecture:

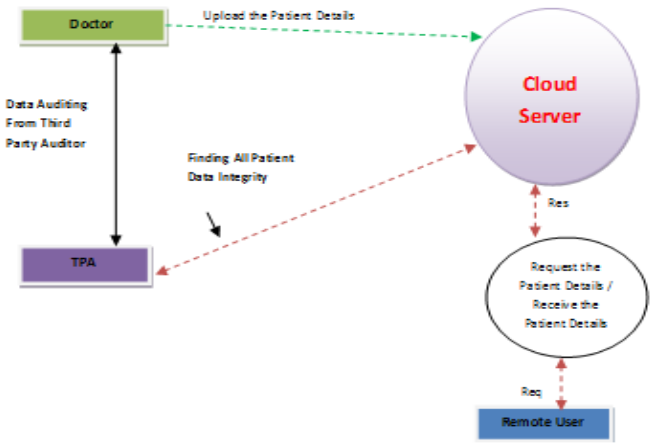

Figure. 1 System Architecture

The Fig 1 shows the overall working of the project. Initially, doctor uploads files in the form of attributes to the cloud server and provides them with keys. Cloud server can verify if any data is modified then provides authorization to access the data by providing the secret key. Then the remote users can access the data.

B. High Level Data Flow Diagrams (DFD):

The Fig 2 shows the flow of data through the system. Initially doctor uploads patients details in the form of attributes to the cloud server and it also assigns keys to remote users to access the data from the cloud server.

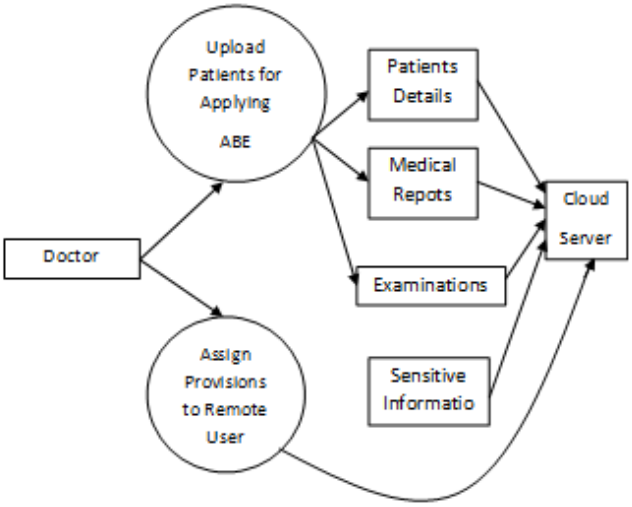

Figure.2 Data Flow Diagram

C. Sequence Diagram:

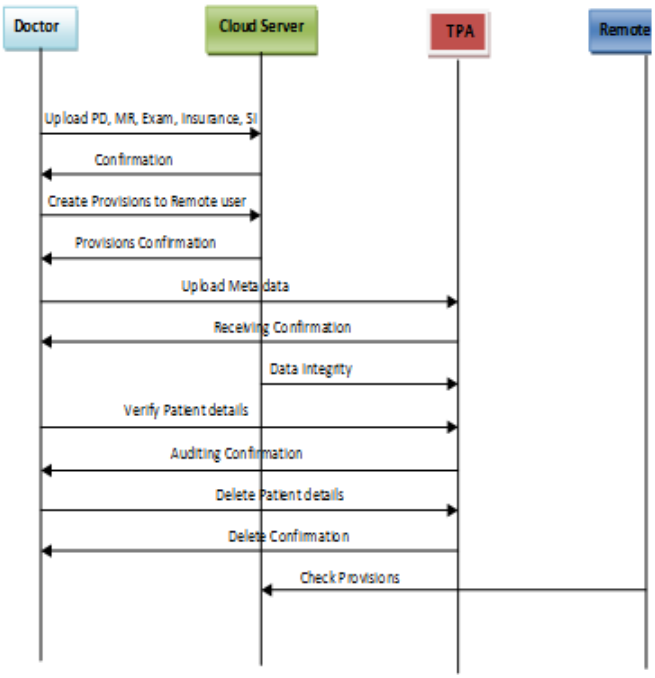

Figure. 3 Sequence Diagram for uploading patient details.

The Fig 3 shows the communication between different modules in terms of sequence of messages.

D. Flowchart:

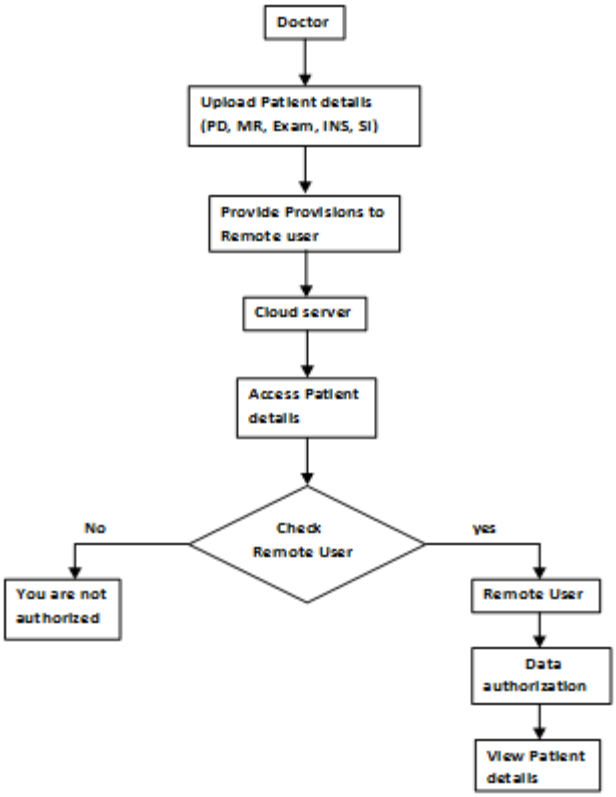

Figure. 4 Flow chart

The Fig 4 shows the flow of operations being carried out from doctor to cloud server and to check the data integrity.

IV. MODULE DESCRIPTION

A. Doctor:

Doctor registers patient details and uploads the patient details to cloud server. Also a secret key will be generated for each file. If the data stored in cloud server is modified, doctor deletes the data from the cloud, re-encrypts the local data and re uploads the data to the cloud server. It sends metadata of different categories to TPA (third party auditor). It provides policy to remote users such as friend, relative and patient. It verifies the data which is stored in the cloud by requesting to the TPA.

B. Cloud Server:

Cloud Server stores the data sent by the data owner, along with the secret key. Authorized person can view the file stored in cloud server. If the data stored in cloud server is modified, it can be known and updated. The TPA can obtain the required data from the cloud server by providing proper authentication.

C. TPA (Third Party Auditor):

It monitors the operation for doctor's data which is stored in clouds. If any data integrity happens in cloud it will give an alert message to the doctor.

D. Receiver/Remote user:

It views the patient details by requesting to cloud server using patient ID and by providing a secret key.

V. CONCLUSION

In this paper, we have proposed a framework of secure sharing of personal health records in cloud computing. Considering partially trustworthy cloud servers, we 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. We adopt an efficient algorithm RSA-128 bit to encrypt the PHR to enhance the security and we use mobile cloud for medical data sharing. Through implementation and simulation, we make our system secure, scalable and efficient.

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]. 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.

[3]. "At risk of exposure – in the push for electronic medical records, concern is growing about how well privacy can be safeguarded," 2006.

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

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

[6]. 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.

[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".

[8]. 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.