RESEARCH ARTICLE OPEN ACCESS

Sharing Of Personal Health Records in Cloud Computing

Rakesh. B[1], Harsha Vardhan. A[2]
[1](M.Tech in CSE Dept, SR ENGINEERING COLLEGE Warangal, ANDHRA PRADESH, INDIA)
[2](Assistant Professor in CSE Dept, SR ENGINEERING COLLEGE Warangal, ANDHRA PRADESH, INDIA)

Abstract
A personal health record, or PHR, is a health record where health data and information related to the care of a patient is maintained by the patient. This stands in contrast to the more widely used electronic medical record, which is operated by institutions (such as hospitals) and contains data entered by clinicians or billing data to support insurance claims. The intention of a PHR is to provide a complete and accurate summary of an individual's medical history which is accessible online. The health data on a PHR might include patient-reported outcome data, lab results, and data from devices such as wireless electronic weighing scales or collected passively from a Smartphone. 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. In Attribute-Based Encryption the decryption of a cipher text is possible only if the set of attributes of the user key matches the attributes of the cipher text. A crucial security feature of Attribute-Based Encryption is collusion-resistance: An adversary that holds multiple keys should only be able to access data if at least one individual key grants access.

Keywords: Attribute-based access control, Auxiliary attribute authorities, Electronic health records, Role-based access control, Security domains.

I. INTRODUCTION

The term "personal health record" is not new. The earliest mention of the term was in an article indexed by PubMed dated June 1978,[2] and even earlier in 1956 reference is made to a personal health log.[3] However, most scientific articles written about PHRs have been published since 2000.

The term "PHR" has been applied to both paper-based and computerized systems; current usage usually implies an electronic application used to collect and store health data. In recent years, several formal definitions of the term have been proposed by various organizations.[4][5][6]

It is important to note that PHRs are not the same as electronic health records (EHRs). The latter are software systems designed for use by health care providers. Like the data recorded in paper-based medical records, the data in EHRs are legally mandated notes on the care provided by clinicians to patients. There is no legal mandate that compels a consumer or patient to store her personal health information in a PHR.

PHRs can contain a diverse range of data, including but not limited to: allergies and adverse drug reactions, chronic diseases, family history, illnesses and hospitalizations, imaging reports (e.g. X-ray), laboratory test results, medications and dosing, prescription record, surgeries and other procedures, vaccinations and Observations of Daily Living (ODLs)

There are two methods by which data can arrive in a PHR.[1] A patient may enter it directly, either by typing into fields or uploading/transmitting data from a file or another website. The second is when the PHR is tethered to an electronic health record, which automatically updates the PHR. Not all PHRs have the same capabilities, and individual PHRs may support one or all of these methods.[1]

In addition to storing an individual's personal health information, some PHRs provide added-value services such as drug-drug interaction checking, electronic messaging between patients and providers, managing appointments, and reminders.[7]

In this paper, we propose a novel and practical framework for fine-grained data access control to PHR data in cloud computing environments, under multiowner settings. To ensure that each owner has full control over her PHR data, we leverage attribute-based encryption (ABE) as the encryption primitive, and each owner generates her own set of ABE keys. In this way, a patient can selectively share her PHR among a set of users by encrypting the file according to a set of attributes, and her encryption and user management complexity is linear to the number of attributes rather than the number of authorized users in the system.

To avoid from high key management complexity for each owner and user, we divide the system into multiple security domains (SDs), where each of them is associated with a subset of all the users. Each owner and the users having personal connections to her belong to a personal domain, while for each public domain we rely on multiple auxiliary attribute authorities (AA) to manage its users and attributes. Each AA distributive governs a disjoint subset of attributes, while none of them alone is able to control the security of the whole system.

In addition, we discuss methods for enabling efficient and on-demand revocation of users or attributes, and break-glass access under emergence scenarios.

1.1. Our Contributions

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.

Symmetric key cryptography (SKC) based solutions. Vimercatiet.al. proposed a solution for securing outsourced data on semi-trusted servers based on symmetric key derivation methods [13], which can achieve fine-grained access control. Unfortunately, the complexities of file creation and user grant/revocation operations are linear to the number of authorized users, which is less scalable. In [4], files in a PHR are organized by hierarchical categories in order to make key distribution more efficient. However, user revocation is not supported. In [6], an owner's data is encrypted block-by-block, and a binary key tree is constructed over the block keys to reduce the number of keys given to each user.

II. RELATED WORK

In PHR system model, 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 shewants 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).

Key escrow (also known as a "fair" cryptosystem) is an arrangement in which the keys needed to decrypt encrypted data are held in escrow so that, under certain circumstances, an authorized third party may gain access to those keys. These third parties may include businesses, who may want access to employees' private communications, or governments, who may wish to be able to view the contents of encrypted communications.

III. PROPOSED WORK

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

3.1. Modules

1. Registration
2. Upload files
3. ABE for Fine-grained Data Access Control
4. Setup and Key Distribution
5. Break-glass

3.1.1. Modules Description

3.1.1.1. Registration

In this module normal registration for the multiple users. There are multiple owners, multiple AAs, and multiple users. The attribute hierarchy of files – leaf nodes is atomic file categories while internal nodes are compound categories. Dark boxes are the categories that a PSD's data readers have access to.

Two ABE systems are involved: for each PSD the revocable KP-ABE scheme is adopted for each PUD, our proposed revocable MA-ABE scheme.

- PUD - *public domains*
- PSD - *personal domains*
- AA - attribute authority
- MA-*ABE* - multi-authority ABE
- KP-ABE - key policy ABE

3.1.1.2. Upload files

In this module, users upload their files with secure key probabilities. The owners upload ABE-encrypted PHR files to the server. Each owner's PHR file encrypted both under a certain fine grained model.

3.1.1.3. ABE for Fine-grained Data Access Control

In this module 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). An attribute-based infrastructure for EHR systems, where each patient's EHR files are encrypted using a broadcast variant of CP-ABE that allows direct revocation. However, the cipher text length grows linearly with the number of un revoked users. In a variant of ABE that allows delegation of access rights is proposed for encrypted EHRs applied cipher text policy ABE (CP-ABE) to manage the sharing of PHRs, and introduced the concept of social/professional domains 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.

3.1.1.4. Setup and Key Distribution

In this module the system first defines a common universe of data attributes shared by every PSD, such as "basic profile", "medical history", "allergies", and "prescriptions". An emergency attribute is also defined for break-glass access.

Each PHR owner's client application generates its corresponding public/master keys. The public keys can be published via user's profile in an online healthcare social-network (HSN)

There are two ways for distributing secret keys.

- First, when first using the PHR service, a PHR owner can specify the access privilege of a data reader in her PSD, and let her application generate and distribute corresponding key to the latter, in a way resembling invitations in GoogleDoc.
- Second, a reader in PSD could obtain the secret key by sending a request (indicating which types of files she wants to access) to the PHR owner via HSN, and the owner will grant her a subset of requested data types. Based on that, the policy engine of the application automatically derives an access structure, and runs keygen of KP-ABE to generate the user secret key that embeds her access structure.

3.1.1.5. Break-glass module

In this module when an emergency happens, the regular access policies may no longer be applicable. To handle 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 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. After the emergency is over, the patient can revoke the emergent access via the ED.

IV. PROBLEM STATEMENT AND ASSUMPTIONS

1.1 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. IMPLEMENTATION

The implementation environment has software such as ASP.NET in Windows XP operating system. The system uses ASP.NET with C# and SQL server 2005

The Login Screen provides the login for the new user and the already existing user. Existing user can login directly by entering the username and the password. If he is a new user then he has to register.

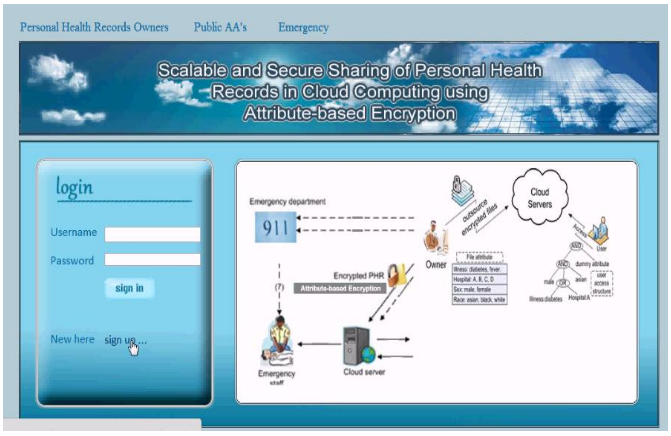
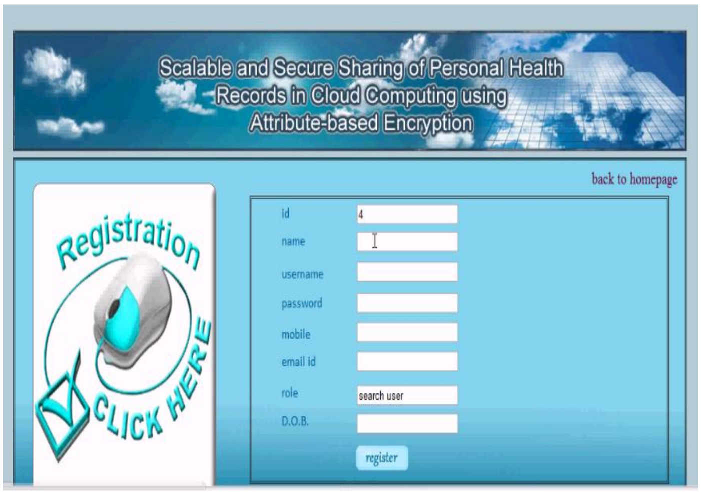


Fig.1 screen shot for user login

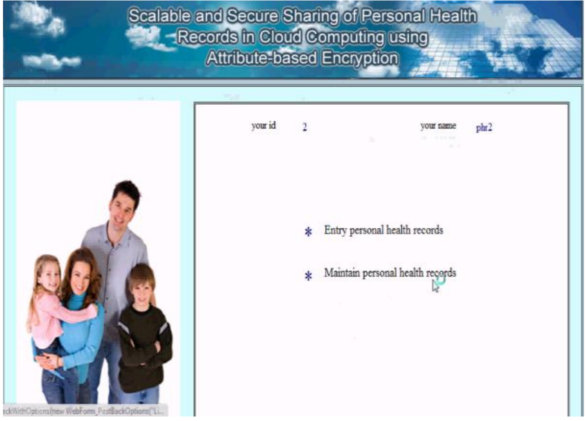
For the registration the user has to enter the id, name, username, password, mobile name, email id and date of birth.


Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption
back to homepage
Registration
CLICK HERE
register

id	4
name	
username	
password	
mobile	
email id	
role	search user
D.O.B.	

Fig .2 Register Page

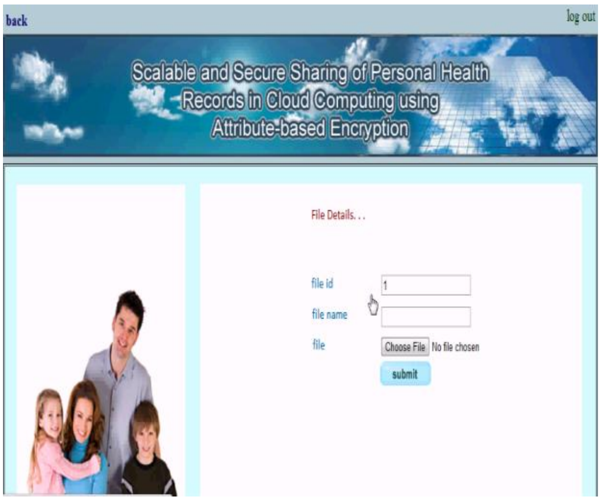
After the successful registration the user gets the symmetric key and the public key.


Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption
* Entry personal health records
* Maintain personal health records
ackWithOptions(new WebForm_PostBackOptions("Li...

your id 2 your name phr2

Fig.3 admin successful login

Admin has the following options enter personal health records and maintain personal health records.



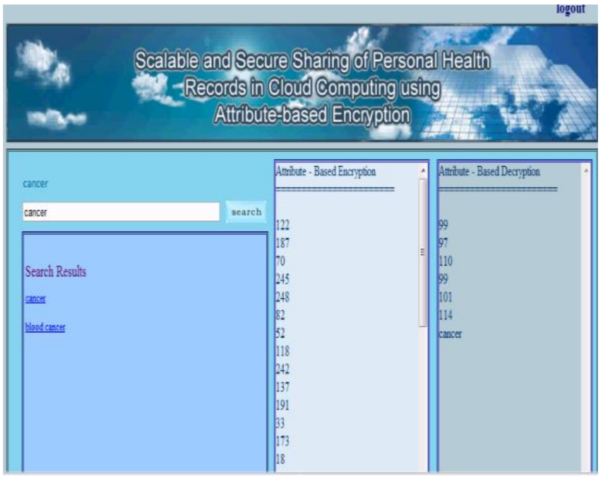
Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption

File Details. . .

file id: 1
file name:
file: Choose File No file chosen

Fig.4 File Upload

Files can be uploaded by file id, file name and select the file to upload. Clicks submit to upload a file.



Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption

cancer

cancer

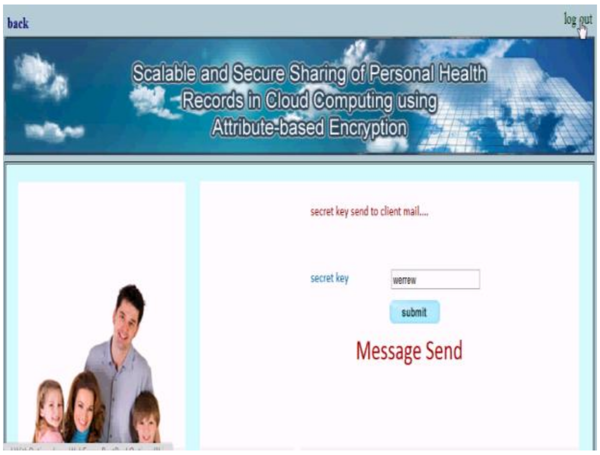
Search Results
cancer
blood cancer

Attribute - Based Encryption
122
187
70
245
248
82
52
118
242
137
191
33
173
18

Attribute - Based Decryption
99
97
110
99
101
114
cancer

Fig.5 Search file by user

Enter the keyword for the disease and click search to search the records corresponding to the disease. To download the file the user need to enter the symmetric key. If the symmetric key entered is wrong then the user is blocked.



Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption

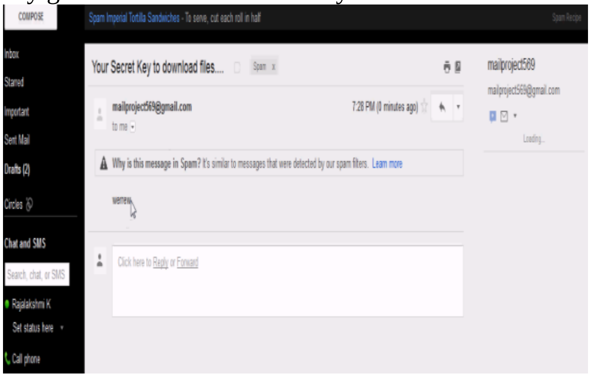
secret key send to client mail....

secret key: werrew

Message Send

Fig.6 Secret Key to the Mail

When the user needs the secret key he has to send a request by verifying the email. After the secret key generation it is send directly to the mail.



Spam Imperial Tortilla Sandwiches - To serve, cut each roll in half

Your Secret Key to download files....

mailproject569@gmail.com
to me
7:28 PM (0 minutes ago)

Why is this message in Spam? It's similar to messages that were detected by our spam filters. Learn more

werrew

Click here to Reply or Forward

mailproject569
mailproject569@gmail.com

Fig.7 Secret Key to Mail

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. An important future work will be enhancing the MA-ABE scheme to support more expressive owner-defined access policies.

REFERENCES

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee,G.,Patterson, D.A., Rabkin, A., Stoica, I., Zaharia, M.: *Above the clouds: A Berkeley view of cloud computing* (February 2009)

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

[3] The health insurance portability and accountability act of 1996 (1996), *http://www.cms.hhs.gov/HIPAAGenInfo/01_ Overview.asp*

[4] 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)

[5] 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)

[6] Wang,W., Li, Z., Owens, R., Bhargava, B.: *Secure and efficient access to outsourceddata. In: CCSW 2009, pp. 55–66* (2009)106 M. Li et al.

[7] Damiani, E., di Vimercati, S.D.C., Foresti, S., Jajodia, S., Paraboschi, S., Samarati,P.*: Key management for multi-user encrypted databases. In: StorageSS 2005, pp.74–83* (2005)

[8] Atallah, M.J., Frikken, K.B., Blanton, M.: *Dynamic and efficient key managementfor access hierarchies. In: CCS 2005, pp. 190–202* (2005)

[9] Blundo, C., Cimato, S., De Capitani di Vimercati, S., De Santis, A., Foresti, S.,Paraboschi, S., Samarati, P.*: Managing key hierarchies for access control enforcement:Heuristic approaches. In: Computers & Security* (2010) (to appear)

[10] Scholl, M., Stine, K., Lin, K., Steinberg, D.: *Draft security architecture designprocess for health information exchanges (HIEs). Report, NIST* (2009)

[11] Ferraiolo, D.F., Sandhu, R., Gavrila, S., Kuhn, D.R., Chandramouli, R.: *ProposedNIST standard for role-based access control. ACM TISSEC 4(3)*, 224–274 (2001)

[12] Jin, J., Ahn, G.-J., Hu, H., Covington, M.J., Zhang, X.: *Patient-centric authorizationframework for sharing electronic health records. In: SACMAT 2009,* pp.125–134 (2009)

[13] di Vimercati, S.D.C., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P.*: Overencryption:management of access control evolution on outsourced data. In: VLDB2007, pp. 123–134* (2007)

[14] Dong, C., Russello, G., Dulay, N.*: Shared and searchable encrypted data for untrustedservers. In: DBSec 2008, pp. 127–143* (2008)

B.Rakesh received the B.Tech Degree in Computer Science & Engineering from Balaji Institute of Technology and Sciences, A.P, India. Currently doing M.tech in Computer Science & Engineering at SR Engineering College, Warangal, India. His research interests include Networking and Security.

A Harsha Vardhan Assistant Professor in the department Computer Science & Engineering, SR Engineering College, Warangal, India.