ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

Attribute Based Encryption for Scalable and Secure Sharing of Medical Records in Cloud Computing Design and Implementation

Bharti Ratan Madnani[1], Sreedevi N[2]

M.Tech Scholar, Dept. Of CS, MVJ College of Engineering, Banglore, India[1]

HOD & Assistant Professor, Dept. Of Computer science, MVJ College of Engineering, Banglore, India[2]

Abstract: Cloud Computing servers provides promising platform for storage of data. Sharing of personal medical records is an emerging patient centric model of health information exchange, which is often outsourced to store at third party, such as cloud providers. The confidentiality of the medical records is major problem when patients use commercial cloud servers to store their medical records because it can be view by everyone, to assure the patients' control over access to their own medical records; it is a promising method to encrypt the files before outsourcing and access control should be enforced though cryptography instead of role based access control. There are various other 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. To achieve fine grained and scalable data access control for medical records stored in semi trusted servers, we leverage attribute based encryption (ABE) techniques to encrypt each patient's medical record file. In this paper, we describe a new approach which enables secure storage and controlled sharing of patient's health data. We explore key-policy attribute based encryption and multi-authority attribute based encryption to enforce patient access control policy such that everyone can download the data ,but only authorize user can view the medical records. This project also supports multiple owner scenarios and divides the users in the system into multiple security domains that greatly reduce the key management complexity for owners and users. A high degree of patient privacy is guaranteed by exploiting multi-authority ABE. In this paper we presents the detail design of modules and implementation Packages of the proposed framework.

Keywords*:* Multi-Authority Attribute Based Encryption, Key-Policy Attribute Based Encryption, Secure Sharing

I. INTRODUCTION

Patient centric medical records information exchange is model for the sharing of medical records, which allows patient to create, manage and control his/her medical information in centralized place through the web or cloud. Patient can now share his/her medical records effectively with a wide range of users such as family members, friends and doctors. Cloud Computing made lots of attraction, because of there is provision of storage as service and software as service, by which software service providers can enjoy the virtually infinite and elastic storage and computing resources. As such, the providers are more and more willing to shift their storage and application services into the cloud like Microsoft and Amazon, instead of building specialized data centers, in order to lower their operational cost .While it is exciting to have these services in the cloud for everyone, there are many security and privacy risks which could impede its wide adoption. The main concern is about the privacy of patients' personal health data and who could gain access to the medical records when they are stored in a cloud server. Since patients lose physical control to their own personal health data, directly placing those sensitive data under the control of the servers cannot provide strong privacy assurance at all.

While going for cloud computing storage, the data owner and cloud servers are in two different domains. On one hand, cloud servers are not entitled to access the outsourced data content for data confidentiality; on the other hand, the data resources are not physically under the full control of data owner. Storing personal medical records on the cloud server leads to need of Encryption mechanism to protect the medical health record, before outsourcing to the cloud. To deal with the potential risks of privacy exposure, instead of letting the service providers encrypt patients' data, medical records sharing services should give patients (patient / medical record owners) full control over the selective sharing of their own medical data. To this end, the medical records should be encrypted in addition to traditional access control mechanisms provided by the server [4].We use Java Paring Based Cryptography library (jPBC) [8] for the implementation of KP-ABE and MA-ABE. In this paper, we discussed the design and Implementation detail for the of the proposed framework.

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

II. ATTRIBUTE BASED ENCRYPTION

The concept of ABE was introduced along with another cryptography called fuzzy identity-based encryption (FIBE) [7] by Sahai and Waters. Both schemes are based on bilinear maps (pairing). In ABE system, users' private keys and ciphertext are labelled with sets of descriptive attributes and access policies respectively, and a particular key can decrypt a particular ciphertext only if associated attributes and policy are matched.

A. Key-Policy Attribute-Based Encryption

The key-policy attribute-based encryption (KP-ABE) was first introduced in 2006 by Goyal et al. [2] In this cryptography system, ciphertext are labelled with sets of attributes. Private keys, on the other hand, are associated with access structures A. A private key can only decrypt a ciphertext whose attributes set is authorized set of the private key's access structure. KP-ABE is a cryptography system built upon bilinear map and Linear Secret Sharing Schemes.

B. Multi-Authority attribute-Based encryption

In a multi-authority ABE system[9], we have many attribute authorities, and many users. There are also a set of system wide public parameters available to everyone (either created by a distributed protocol between the authorities). A user can choose to go to an attribute authority, prove that it is entitled to some of the attributes handled by that authority, and request the corresponding decryption keys. The authority will run the attribute key generation algorithm, and return the result to the user. Any party can also choose to encrypt a message, in which case he uses the public parameters together with an attribute set of his choice to form the ciphertext. Any user who has decryption keys corresponding to an appropriate attribute set can use them for decryption.

III. RELATED WORK

For access control of outsourced data, partially trusted servers are often assumed. With cryptographic techniques, the goal is trying to enforce who has (read) access to which parts of a patient's PHR documents in a fine-grained way.

A. Symmetric key cryptography (SKC) based solutions:

Symmetric-key algorithms are a class of algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of ciphertext. The keys may be identical or there may be a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link Vimercati et.al.[6] Proposed a solution for securing outsourced data on semi-trusted servers based on symmetric key derivation methods , 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.

B. Public key cryptography (PKC) based solutions:

PKC based solutions were proposed due to its ability to separate write and read privileges. To realize fine-grained access control, the traditional public key encryption (PKE) based schemes proposed by J. Benaloh, M. Chase, E. Horvitz, and K. Lauter [1] in their work "Patient controlled encryption: ensuring privacy of electronic medical records" ,they purpose the solution scenario and shows how public and symmetric based encryption used , disadvantage of their solution is either incur high key management overhead, or require encrypting multiple copies of a file using different users' keys.

C. Attribute Based Encryption based solutions:

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). Narayan et al. proposed an attribute-based infrastructure for EHR systems, where each patient's EHR files are encrypted using a broadcast variant of Cipher Text-ABE (CP-ABE)[4] .However, the ciphertext length grows linearly with the number of unrevoked users. In [16], a variant of ABE that allows delegation of access rights is proposed for encrypted EHRs. Ibraimi et.al. [5] applied ciphertext policy ABE (CP-ABE) [18] to manage the sharing of PHRs, and introduced the concept of social/professional domains but they do not use multi-authority ABE . In [3], 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. Drawback is device dependency and revocation is not supported. Other Common drawback of all above solutions is problem of key-escrow as they consider single trusted authority.

IV. PROPOSED SOLUTION

A. Architecture :

Fig.1 depicts the architecture of proposed system for secure sharing of the medical records. The system is split into two security domains namely, public domains (PUDs) and personal domains (PSDs) according to the different

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

users' data access requirements. The PUDs consist of users who make access based on their professional roles, such as doctors, nurses, medical researchers and insurance agents. For each PSD, its users are personally associated with a data owner (such as family members or close friends), and they make accesses to medical records based on access rights assigned by the owner. Here we consider Data owner who possess the medical record, data reader as who can read the encrypted medical record. In PSD, the owner used key-policy attributed based encryption and generates secret key for their PSD users and in PUD the multi-authority attribute based encryption is preferred. Secret Key for PUD users are generated by Multiple authority (For this paper we consider Specialization Authority and Medical Authority) depending on their specialization and profession in combine.

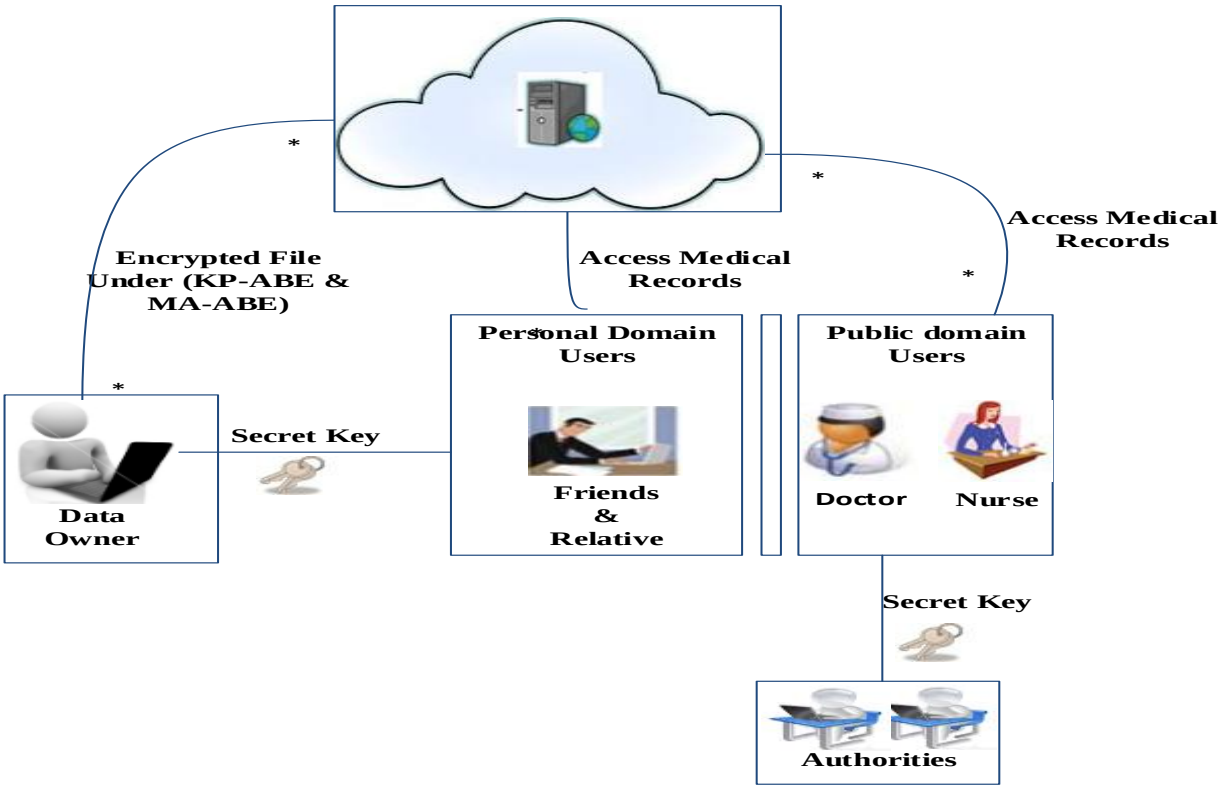

Fig-1 Architecture of Patient Medical Record Sharing

B. Design of Modules:

The operations of proposed medical record sharing system combine KP-ABE and Multi-Authority ABE and traditional cryptography, allowing patients to share their medical records. These operations can be classified into following modules: In this section we discuss main module design concept for sharing of medical records using Attribute based encryption – (KP-ABE and Multi Authority-ABE).

Modules of the system are:

1. System Setup and Secret Key Generation
2. Encryption of Medical Records
3. View Medical Records (Decryption)
4. Revocation Of Public domain User / attributes

1) System Set-Up and Key-Generation : As system is divided into two domain , both domains has different procedure for Set-up and Key Generation. In Set-Up public and master parameters are generated, which are , used for key generation, encryption and decryption.

a) Personal Domain :

The system first defines a common universe of data attributes shared by every PSD, such as "personal info", "medial history", "allergies", and "prescriptions" "emergency" , "friend", "relative" , "emergency". An emergency attribute is also defined for break-glass access. Each data owner's client application generates its corresponding

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

public/master keys using Key-Policy attribute Based Encryption. The public keys can be published with help of system provided by service provider.

Data Owner specify the access policy of data reader in her personal domain, and generates secret key using Key-Policy attribute Based Encryption. Personal domain user obtains secret key from the data owner through secure email by sending a request for the keys. or data owner send the secret key to personal domain user via secure email. Example of Policy has the following form in the postfix format:

"Personal-medical-record personal-Information or family and"

Fig -2,3 and 4 shows the use case diagram and sequence diagram for set-up and key generation.

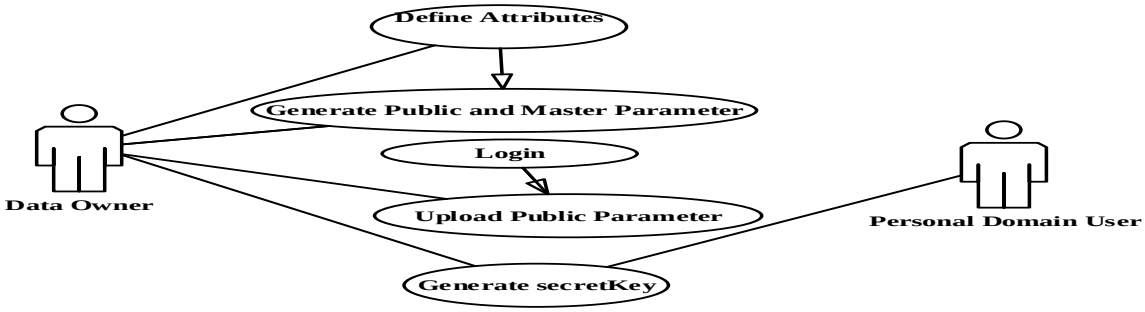

Fig. 2 Use case Diagram for setup and Key Generation (Personal Domain)

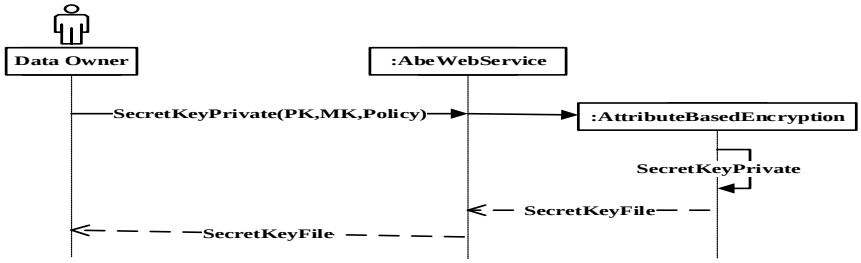

Fig. 3 - Sequence Diagram for the Set-Up for Personal Domain User

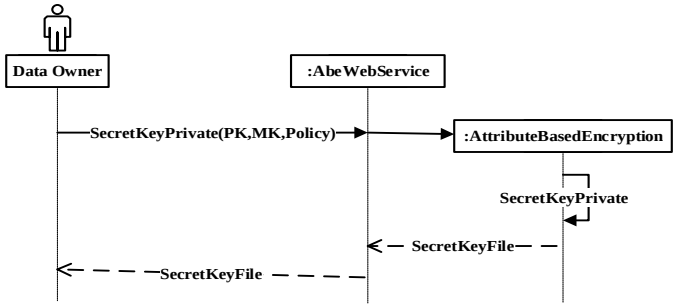

Fig. - 4 Sequence Diagram for the secret key for Personal Domain User

b) Public Domain:

The system defines role attributes, and a reader in a public domain obtains secret key from AAs, which binds the user to her claimed attributes/roles. For example, a physician in it would receive "physician", "internal medicine" as her attributes from the Medical Authority and Specialization Authority respectively. In practice, there exist multiple AAs each governing a different subset of role attributes. AA in combine generates Global public parameter and attributes specific public and master parameter of their respective attributes using MA-ABE Setup discuss in next section. And publish public parameters with help of service provider. Two authorities Medical and Specialization are considered for this paper. Medical Authority monitors professional attributes for example "physician, Doctor , Nurse , Pharmist" and Specialization Authority monitors Specialization of PUD for example "Internal Medicine , EndoDentist, Surgery ".

All Authority in combine generates the secret key for the public domain user of their claimed role attributes and send via secure email or in person public domain user has to obtain the secret key. Fig – 5 shows the use case diagram for Set-Up and Key Generation

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

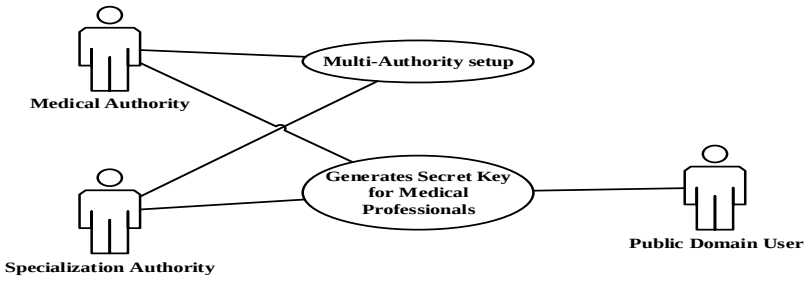

Fig 5– Use case Diagram for Set-Up and Key Generation (Public Domain)

2. Encryption: The Patient Encrypt the medical records under a certain fine grained and role-based access policy for users from the Public domain to access, and under a selected set of data attributes that allows access from users in the Personal. And Uploads Encrypted File to the server. Detail of the Encryption process is discussed in next section.

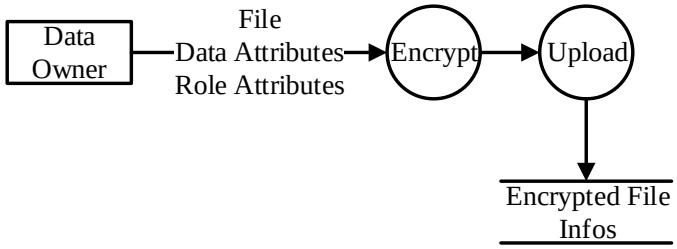

Fig 6– Data Flow Diagram for Encryption and Upload

3. View Medical Record File /Decryption: User from the personal or public domain can request the file form the server. Only user can view the records, provided the secret key policy matches with the at tributes attached with the files. Fig below shows the Flow Diagram for the decryption.

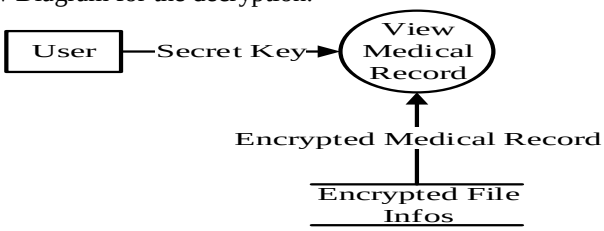

Fig 7 – Dataflow Diagram for the Decryption

4 Revocations:

Here we consider the revocation public domain users attributes. Revocation of user is similar to revocation of all attributes of the user. The Revocation of user attribute is done using following steps:

1. Attribute Authority redefines the MK and PK of the attributes of the revoked user and also generates re-encryption and re-secret keys for files and secrets key respectively
2. Attribute Authority sends the PRE keys for secret key to unrevoked user via secure email to public domain user and public domain user updates the secret key using re-secret Secret Keys.
3. Authority re-encrypts the encrypted medical files stored on server using proxy re-encryption key generated in step1.

Fig 8 shows the use case diagram for the revocation attributes of user of the public domain.

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

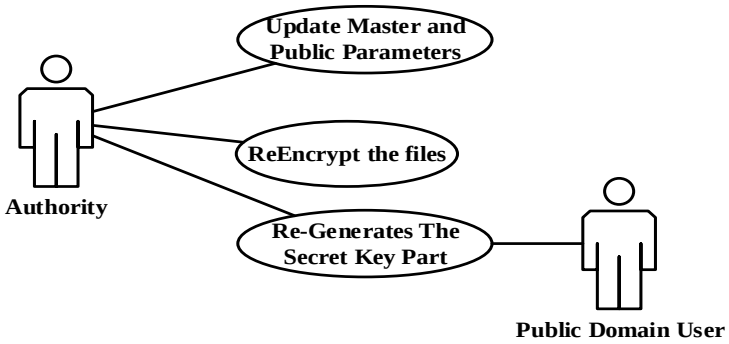

Fig 8 Use case diagram for the Revocation

C. System Implementation:

This stage focuses on specific tools such as programming languages, libraries and components which allow to quickly producing software of high quality. 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.

1) *Main Software Requirements* are as follows:

1. java Pairing Based Cryptography[8] – library and Net Bean-IDE for KP-ABE and MultiAuthority API development, SQL Server ,
2. SQL Server and Microsoft visual studio 2012 for GUI development

2) *Main Algorithms*

Algorithms of for KP-ABE with enhancement are discussed as below:

1) KP-ABE Setup(A): Outputs public key PK and Master key MK for A as set of attributes

- Associate for each attribute in A with attributes universe as U = {1, 2, ..., n}.
- It defines a bilinear group G1 of prime order p with a generator g, a bilinear map e : G1 × G1 -> G2 which has the properties of bilinearity, computability, and non-degeneracy.

-Associate each attribute i ∈ U with a number ti and also chose y uniformly at random in Zp* and y.

-The public key is:

$PK = (T_1 = g^{t1}, ..., T_1 = g^{t|U|}, Y = e(g, g)^y)$

-The master key is:

$MK = (t_1, \ldots, t_{|U|}, y)$

2) KP-ABE Encryption (M, γ, PK) :M message in G_T with a set of attributes γ , PK is public Key , outputs CipherText E.

-Choose a random value s in Zp. Encrypt a secret message M in G_T with a set of attributes γ.

-The ciphertext is:

$E = (\gamma, E' = MY^s, \{E_i = Ti^s\}$ where $i \in \gamma)$

3) KP-ABE Key Generation (A, MK) :

This algorithm output a secret key D embedded with a access structure T. The access structure A is realized by the following three steps:

1. For root node r, set value secret = y. mark all node un-assigned and mark root node assigned.
2. Recursively, for each assigned non-leaf node,
   a. If the operator is ^ (and) and its child nodes are marked un-assigned, let n be the number of child nodes, set the value of each child node, except the last one, to be $s_i \in Zp$, and the value of the last node to be $s_n = s - \sum s_i$ .Mark this node assigned.
   b. If the operator is v (or), set the values of its child nodes to be s. Mark this node assigned.
3. For each leaf attribute $a_{j,i} \in T$, compute $D_{j;i} = Tj^{\wedge}s_i$

Output: Secret Key Sk= { $D_{j,i}$}

4) KP-ABE Decryption (E, D) This algorithm takes as input the cipher text E encrypted under the attribute set U, the user's secret key SK for access tree T, and the public key PK. Finally it output the message M if and only if U satisfies T.

Basic Algorithm of the MA-ABE with enhancement is:

1) KeyIssue(Attributes ,MK, PK). This algorithm, the AAs collectively actively generates a secret key for a user. For a user with (secret) ID *u*, the secret key is in the form :

$SK_u = \langle Du = g^{\wedge}R_u, \{D_{k;i} = g^{\wedge}(q_k(i)/t_{k,i}), Ver_{k,i}\}_{k \in \{1 \ldots N\}} \rangle$

where *Ru* is a global ID for user *u*, and $q_k(0) = \Sigma_k v_k - R_u$.

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

2) Encryption(M,PK,attributes[]): This algorithm takes a message *M*, *PK* and a set of attributes and outputs the ciphertext *E* as follows:
The encryptor first chooses an $s \in Zp$, and then returns :
$CT = [E0 = M \cdot Y^{\wedge} s, E1 = g_2{\wedge}s, \{C_{k;i} = T_{k;i}{\wedge}s, Ver_{k;i}\}_{;k\in\{1..N_i\}}]$
where i= no of attributes form authority k

3) Decryption(*CT, SKu*): This algorithm takes as input a ciphertext *CT* and a user secret key *SKu*. If for each AA *k*, If the version of attribute in SK and CT matches, algorithm pairs up $D_{k;i}$ and $C_{k;i}$ and reconstructs $e(g1, g2)^{\wedge}sq_k(0)$. After multiplying all these values together with $e(Du,E_1)$, *u* recovers the blind factor $Y^{\wedge}s$ and thus gets *M*.

4) Update Parameter : This algorithm updates an attribute to a new version by redefining its system master key and public key component. It also outputs a proxy re-encryption key and re-secret-key between the old version and the new version of the attribute.

5) UpadteSecretKey : This algorithm translates the secret key component of attribute i in the user secret key SK from an old version into the latest version using re-secret-key generated in step 4.

6) ReEncryptFile : This algorithm translates the ciphertext component of an attribute i of a file from an old version into the latest version using proxy- encryption key generated in step 4.

3) *Main Packages and Systems* **:**

a. *Package kpabe :* Implements Key-Policy Attribute-Based Encryption Algorithms such as Setup, Encryption, Key Generation, and Decryption, Policy Generation algorithms, policy update. They are the most important parts during KP-ABE construction.

b. *Package MultiAuthority* : implements Multi-Authority Attribute-Based Encryption Algorithms such as Setup: implements Setup, Encryption, Key Generation, and Decryption, Policy Generation algorithms , policy update. Update Parameter , Regenertate secret key and reEncrypt the files.

c. *Package AbeEncrypt:* implements the Encrypt – which encrypt the file under both data attribute under key-policy ABE and MultiAuthority-Based Encryption. Main functions are DecrytMedicalRecordPrivate, DecrytMedicalRecordPublic and Encrypt.

d. *Package AbeWebService:* Provides the interface with above packages, which can be used in any platform to generate GUI using Web Service. Using this package , it makes easier to use API for any platform.

e. *PMRS System:* Allows user to register with system and encrypt medical records, upload medical records, view medical records using above packages.

4) Screen Shots : Here are the two main screen shots of the project.

a) Encryption of medical records: fig 9 below shows the screen shot for the Encryption.

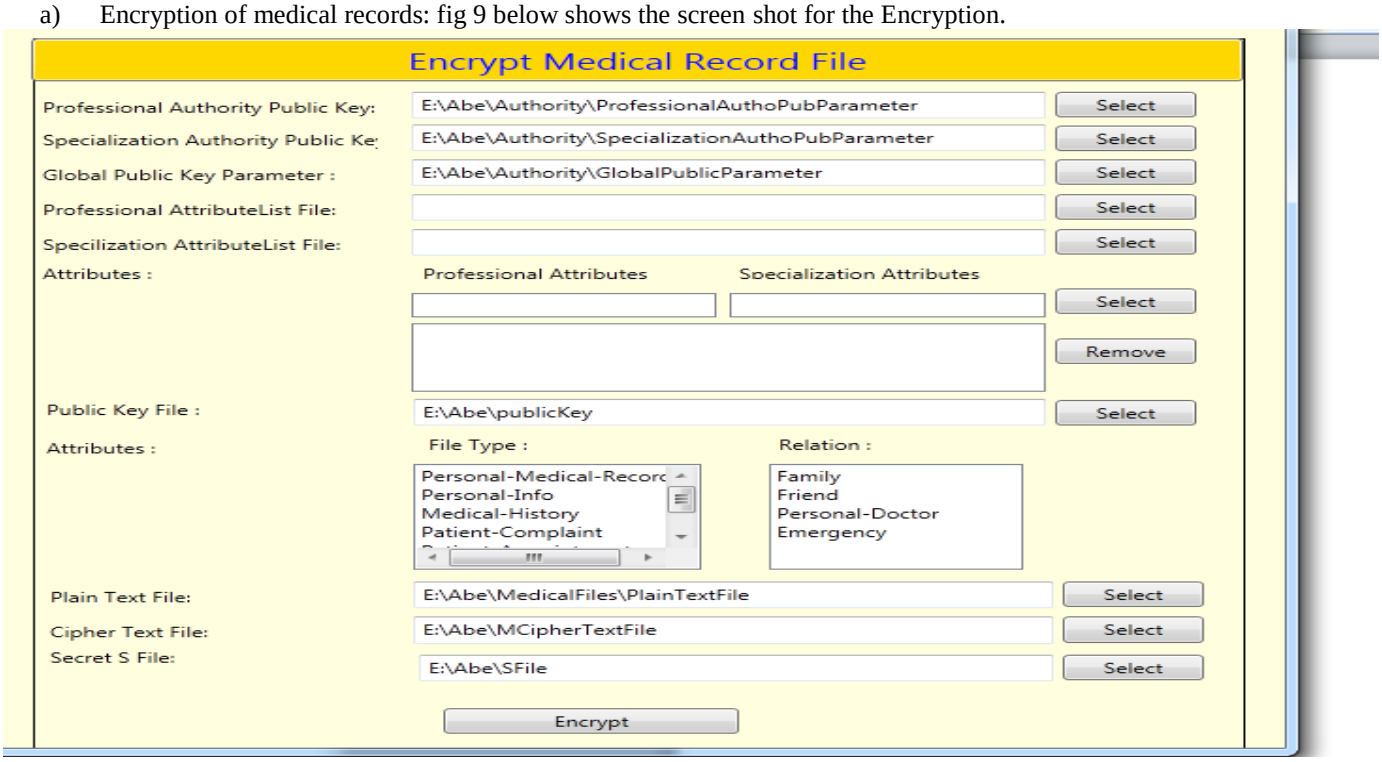

Fig- 9 Encryption Of Medical Records

ISSN (Print) : 2320 – 9798
ISSN (Online): 2320 – 9801

International Journal of Innovative Research in Computer and Communication Engineering
Vol. 1, Issue 3, May 2013

Data Owner selects the attributes from public and personal domain , select the plain text file which is the medical file wants to encrypt , also provides the public parameter from both public domain – authorities global parameter and public parameter as input. Two files are generated as output cipher text file – which is encrypted medical records and Sfile – which contains the secret s used for the encryption.

Encrypted medical record file is uploaded by data owner to the PMRS system.

b) Decryption: Fig 10 below shows the screen for the Decryption – View Medical Records.

Uploaded medical record can view by the personal domain or public domain user , by providing secret key matches with the attribute list of the encrypted file.

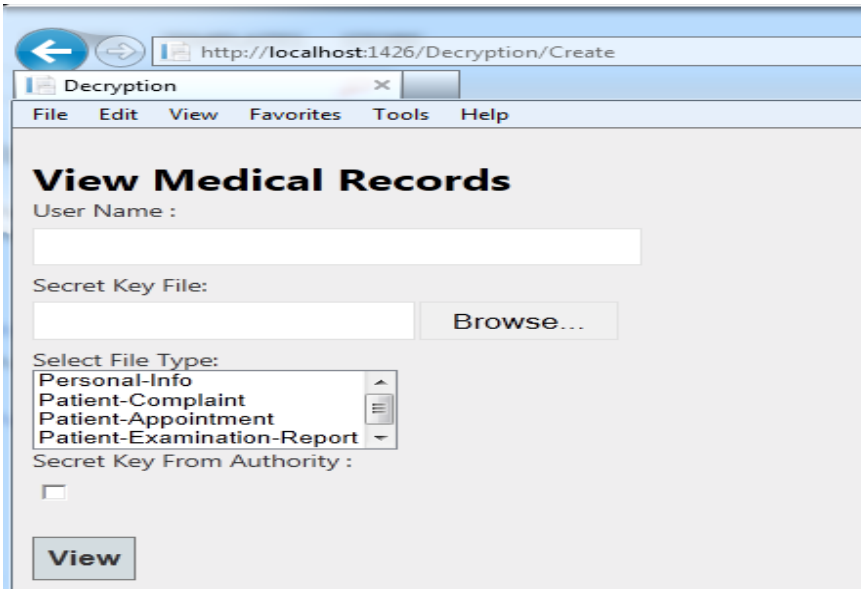

Fig 10 Screen shot for the Decryption – View Medical Records.

V. CONCLUSION

In this Paper, we have presented the detail design and implementation detail of proposed a novel framework of secure sharing of personal medical 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 medical record files to allow fine-grained access. The framework addresses the unique challenges brought by multiple owners and users, in that we greatly reduce the complexity of key management while ensured the privacy. We utilize various forms of ABE to encrypt the medical record files, 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.

REFERENCES

[1]. 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.

[2]. 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.

[3]. 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 on mobile device. Technical report, Cryptology ePrint Archive, Report 2010/565, 2010. http://eprint. iacr. org/2010/565.

[4]. S. Narayan, M. Gagn´e, and R. Safavi-Naini, "Privacy preserving ehr system using attribute-based infrastructure," ser. CCSW '10, 2010, pp. 47–52.

[5]. L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker, "Ciphertext-policy attribute-based threshold decryption with flexible delegation and revocation of user attributes," 2009.

[6]. S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-encryption: management of access control evolution on outsourced data," in VLDB '07, 2007, pp. 123–134.

[7]. A. Sahai and B. Waters. Fuzzy identity-based encryption. Advances in Cryptology {EUROCRYPT 2005, pages 457{473, 2005.

[8]. Angelo De Caro and Vincenzo Iovino, "jPBC: Java Pairing Based Cryptography" Computers and Communications (ISCC), 2011 IEEE Symposium on Digital Object Identifier: 10.1109/ISCC.2011.5983948 Publication Year: 2011 , Page(s): 850 – 855

[9]. M. Chase and S. S. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in CCS '09, 2009, pp. 121–130.