Secure Cloud based Medical Data exchange using Attribute based Encryption

Shini S G
Department of Computer Science & engineering
SCT College of Engineering
Trivandrum, India

Chitharanjan K
Department of Computer Science & engineering
SCT College of Engineering
Trivandrum, India

ABSTRACT
Secure Management of medical data has become a major issue as there is an increase in need for medical data exchange among different healthcare providers. Cloud platform can form an exchange platform that all healthcare organizations use and can serve as storage centre of medical records. However, there had been wide security and privacy concerns as medical records are known to third–party server and unauthorized parties. The medical data residing on a cloud server are subjected to many inside and outside malicious attacks. To keep sensitive medical data confidential in cloud, existing solutions apply encryption methods by disclosing data decryption keys only to authorized users. Then also issues like risk of information disclosure, user revocation, scalability in key management are present which hinders to achieve fine grained data access control. To achieve fine grained and scalable access control for medical records, attribute based encryption techniques are used to encrypt medical data. The main method is to map an access control policy into a secret encryption key and then to encrypt the data under the encryption key such that only authorized users who possess the decryption key can access the data in cloud. The secret key is associated with a set of attributes which identify the particular user. The user can decrypt the data if and only if his attributes satisfy access control policies. The proposed method supports efficient user revocation and achieves break glass in emergency situations. The proposed scheme is implemented at real time cloud environment in Microsoft Azure.

Keywords
Medical Records,Cloud Computing,Attribute Based Encryption,User Revocation.

1. INTRODUCTION
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. Information sharing across providers is inefficient and data probability is rare. Health information Exchange (HIE) is the provision of 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.

An Electronic Medical Record (EMR) is the effective capture, dissemination, and analysis of medical and health related information for a single patient [18]. All participants in the health care delivery system have a stake in efficient information flows. They include health care providers, insurers, government agencies, claims processors, and patients. Indeed, Electronic Medical Records managed by individuals are termed Personal Health Records (PHRs). PHRs capture all relevant personal health details, including diagnoses, X-Rays, and similar items 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. The main components of an EMR is shown in Fig 1.The electronic medical records only support individual hospitals and do not provide communication or share resources among hospitals. Therefore, it is difficult for a patient to visit his doctor in one hospital and have his medical record from another hospital available. To eliminate this problem, electronic medical data sharing techniques are used. The emerging technologies like peer-to-peer (P2P) systems and cloud computing [14] are capable of enabling sharing of electronic medical records across autonomously managed heterogeneous healthcare information systems.

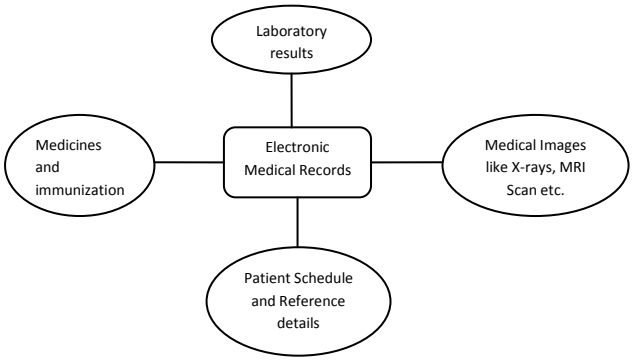

Figure 1: Electronic Medical Record System

Cloud computing [14] technology can simplify the complex medical records exchange procedure between different systems, and save the device setup expenses for smaller hospitals. It can provide an exchange platform that all hospitals and clinics can use, and can serve as electronic medical data storage [1]. Large companies like Google and Microsoft are building medical record clouds such as Google

health and Microsoft Health Vault. Through the health care cloud of the cloud platform, patients need only one interface to find out their complete medical history, instead of having through different hospitals at the risk of finding only a partial medical history. The benefits of putting health data in a cloud based system [4] include:

- **Data portability:** Using Cloud based medical data exchange; it is easier to access and share data between patients and doctors and between specialists.
- **Better security:** Data and medical records are not stored at the normal location. Instead, data is stored in a safe, HIPAA-compliant secure cloud location allowing for convenient, secure access from any location with the benefit of off-site disaster recovery.
- **Enormous storage capacity:** With cloud-based systems, doctors and clinicians do not have to own hardware and software. Additional data storage is available as needed.

Cloud computing inevitably poses new challenging security threats [6] for number of reasons. The use of the cloud for healthcare, including electronic medical records, personal health records, and healthcare treatment expands the protection of personal information based on the Health Insurance Portability and Protection Act (HIPPA).Firstly, existing cryptographic primitives for the purpose of data security protection cannot be directly used because it will loss the user's control of data under Cloud Computing. Therefore, security of correct data storage in the cloud must be conducted without knowing the whole data. Secondly, Cloud Computing cannot be considered as a third party data warehouse. To ensure storage security under dynamic data environment is hence an important matter.

To protect the confidential medical information in cloud, encryption is used. But it requires exchange of decryption keys among the data owners and users. It is not good for the data owner to remain online for providing decryption key to registered users. The solution is to delegate the distribution task to cloud server. The risk of privacy violation increases when decryption keys are distributed via cloud storage provider. The main problem with this solution is the increase in load of asymmetric encryption on the data owner.

For securing medical records stored in cloud and achieving fine grained access control, the proposed scheme combines Key Policy based encryption, along with Proxy Re-encryption.KP-ABE[2] mainly concentrates on access control policy and PRE [12] delegates task of decryption key distribution to cloud server. By uniquely combining these cryptographic techniques this scheme realizes a secure medical record exchange through cloud platform with minimum overhead on data owner.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 outlines system models and assumptions. Section 4 presents the main proposed scheme. Section 5 and 6 provides the performance and security analysis of the proposed scheme. Section 7 presents the computation and communication assessment of KP-ABE in the proposed scheme. Section 8 concludes the paper along with the future directions.

2. RELATED WORKS

Kallahalla et al [5] proposed Plutus as a cryptographic file system to secure remote file stored in untrusted servers. Plutus groups a set of files sharing similar attributes as a file-group and associates each file-group with a symmetric lockbox-key. Each file is encrypted using a unique file-block key which is then encrypted with the lockbox-key of the file group to which the file belongs. If the owner wants to share a file-group, he just gives the corresponding lockbox-key to users. As the complexity of key management is proportional to the total number of file-groups, Plutus is not suitable for achieving fine-grained access control in which the number of possible "file-groups" could be huge.

Ateniese et al [9] proposed a secure distributed storage scheme on remote servers based on proxy re-encryption. The data owner mainly encrypts blocks of content with symmetric content keys. The content keys are all then encrypted with a master public key, which can only be decrypted by the master private key owned by the data owner. The data owner uses his master private key and user's public key to generate proxy re-encryption keys and semi-trusted server use these keys to convert the cipher text into that for a specific granted user and fulfill the task of access control enforcement. The main problem with this scheme is that collusion between a malicious server and any single malicious user would disclose decryption keys of all the encrypted data, which will affect the data security of the system completely. In addition, user's access privilege is not protected from the proxy server. User secret key accountability is also not considered.

Patient Controlled Encryption [10] is a privacy preserving medical health record system. In PCE data is stored as hierarchical structure on a remote server and patient has no direct control on these data. However, PCE facilitates sharing and searching of the encrypted data in the remote location. The proposed scheme can be realized using symmetric and asymmetric encryption algorithms, with their inherited benefits and limitations.

Secure patient-centric access control (PEACE) [10] is a scheme for the emerging electronic health care (eHealth) system. In order to assure the privacy of patient personal health information (PHI), they define different access policies to users according to their roles, and then assign different attribute sets to the data requesters. By using these different sets of attributes, construct the patient-centric access policies of patient PHI. The PEACE scheme can guarantee PHI integrity and confidentiality by using digital signature and pseudo-identity techniques. It uses identity based cryptography to aggregate remote patient PHI securely. Extensive security and performance analyses demonstrate that the PEACE scheme is able to achieve desired security requirements at the cost of an acceptable communication overhead.

Vimercati et al [3] proposed a solution for securing data storage on untrusted servers using key derivation methods [9]. In this proposed scheme, each file is encrypted with a symmetric key and each user has given a secret key. To grant the access privilege for a user, the owner creates corresponding public tokens from which, together with his secret key, the user is able to derive decryption keys of desired files. The owner then transmits these public tokens to the semi-trusted server and delegates the task of token distribution to it.

3. MODELS, DESIGN GOALS AND ASSUMPTIONS

3.1 System Model

The proposed system consists of Data Provider, Data Consumers and Cloud Service provider. Data providers use the storage capacity provided by CSP by uploading the encrypted files for exchange. Data consumers download a

copy of data from cloud server and decrypt it by using his decryption key. Neither data provider nor the user is always online.CSP is always online and has storage capacity and computation power.

3.2 Security Model

In this work, we just consider CSP to be semi trusted, i.e,"honest but curious". That means the cloud server will honestly perform the task delegated by the owner, but they will try to find out as much sensitive information in stored medical data as possible. At the same time, some users will also try to access the files beyond their scope of access privileges. For e.g., Drug companies may want to obtain the prescriptions of patients for understanding the buying patterns and boosting their profits. To do so they may collude with cloud servers for getting beneficial results. The Proposed work focuses on fine grained access control in a cloud based medical data exchange. We believe that a security channel (SSL) is present between the involved entities through data is exchanged.

3.3 Design Goals

Our main goal is to achieve secure patient centric medical access control and secure key management at same time. The system guarantees negligible execution overhead on both the owner and user, while allowing guaranteed user revocation. The proposed method should prevent cloud servers from knowing both data file contents and access privilege information of user.

4. PROPOSED SCHEME

By combining KP-ABE, PRE and lazy–encryption, we leverage patient to ensure fine grained access control over the outsourced data in the cloud. Table 1 explain the notation used in the descriptive detail of the proposed scheme. The schematic description of the proposed scheme is shown in Fig.2.Patient wants to upload his medical details to cloud and the authorized doctor downloads these from cloud for diagnosis purposes. Before uploading, patient sends URL of cloud storage, his secret key and PRE key to the doctor through email. He then encrypts the medical data files with any of the symmetric encryption algorithms. The encryption

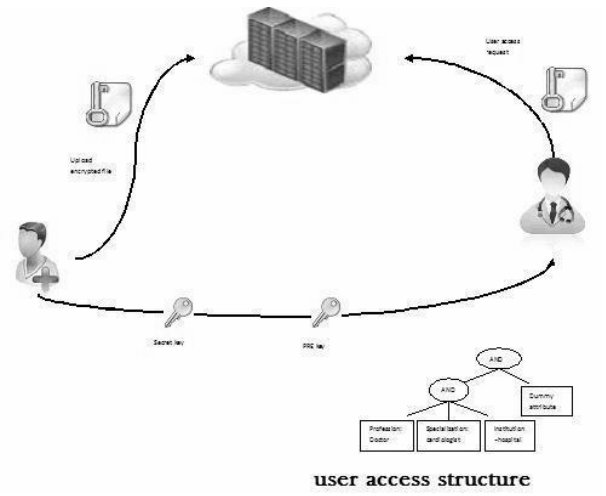

Figure 2: Secure Cloud based Medical Data Exchange

key DEK is again encrypted with KP-ABE which has an access structure that can be satisfied with secret key. Then encrypted files along with DEKs are uploaded to cloud. Then doctor can request files along with the DEK.On receiving response from cloud, doctor use secret key to decrypt the DEK and using DEK he decrypts the encrypted file. Before getting into details of proposed scheme, we mention some of the assumptions taken during its design

1. We assume that users behave honestly, so that they never share their decryption key with revoked users.

2. We believed that cloud server performs his duty honestly given by data owner.

Table 1. Abbreviations Used in main process

Notation	Description
DEK	Data Encryption
PK	Public Key for KP-ABE
MK	Master Key for KP-ABE
PRE	Proxy Re-encryption
P	Access structure
SK	Secret Key
C	Cipher text
UL	User List
S	Size of data file
A	Attribute set
n	Number of users

4.1 Data Preparation

It is the first step in proposed scheme. It encrypts the data such that it can be used to meet the security requirements specified in system design goals. Owner randomly selects the data encryption key DEK from his local key space K.DEK is used to encrypt the sensitive data file with an arbitrary symmetric algorithm. Here we use DES, AES and triple DES for comparative study.

4.2 Initialization

In this operation, data owner takes α as a security parameter and outputs the system public key PK and system master key MK.The data owner signs each component of PK along with these signatures to cloud servers.

4.3 New File Creation

Once File f is encrypted, the next step is to upload it to the cloud based storage along with DEK.To avoid dependency on third party and cloud storage provider, DEK is secured by using attribute based encryption. Data owner associates data file with set of attributes for access structure P, and then encrypts DEK with KP-ABE.P enumerates the attributes which are required for the decryption; these attributes are associated with SK of a legitimate user.

4.4 Key Generation

Key Generation process generates SK to decrypt the medical file. Keys are generated by using MK and P.SK represents

attributes involved in P.Secret attributes are computed to represent the secret value used for encryption process and then during decryption process these attributes are interpolated to reveal the secret value.

4.5 Key Distribution

In this proposed scheme we use two types of keys, DEK and SK.Legitimate user must possess each of them to access the encrypted data in cloud. According to security model DEK is subjected to change on each user revocation. However SK is partially updated on each user revocation.

The encrypted file and Data encryption key is uploaded to cloud server. Secret key and confirmation code is send as email, once patient approve the request. This is only exchange of confidential information between owner and user. Decryption key and secret key are updated via cloud servers utilizing proxy re-encryption.

4.6 New User Registration

When a new user is registered to system, the data owner assigns an access structure and secret to him through following process.

- Select a unique identifier ID and an access structure P for new user
- Generate a secret key SK for W as described in key generation process
- Encrypt P, SK, PK with user ID's public key, denoting the cipher text by C
- Send the encrypted file, signature, DEK and secret key to cloud server.

On receiving the cloud server work as follows

- Verify signature and proceed if correct
- Store new user W in system user list UL
- Forward C to user

On receiving C, the user first decrypts it with his private key. Then he verifies the signature .if correct he accepts (P, SK, and PK) as his access structure, secret key and system public key.

4.7 User Revocation

User revocation is performed whenever there is a need to restrict the user from accessing the outsourced data, which was previously accessible to him. User revocation in [15] is applied for restricting a user when he leaves the system or owner doesn't want him to access his medical data. The main idea is to let owner to update the affected attributes for all the remaining users. The computational overhead on owner increases because data owner has to re-encrypt all data files accessible to the leaving user and for updating secret keys for users. In order to reduce the complexity on owners, proxy re-encryption is used to delegate operations to cloud server along with preserving confidentiality of data. Further reduction of computation overhead on cloud server is achieved by using lazy re-encryption technique. By using traditional proxy re-encryption scheme, owner assigns a re-encryption key to database which can be used to re-encrypts the encrypted medical data into encrypted file with requester's public key. When the database is corrupted, some of the encrypted medical file may be disclosed to unauthorized individuals and using this proxy key he can re-encrypt all encrypted data files. To avoid this problem, type based proxy re-encryption [16] is used for assigning many key pairs to different types of medical data. The data owner categorizes her medical data according to his privacy concerns. For example patient set his medical history as type t_1,his family details as type t_2 etc.For each type of medical files, the data owner assigns a PRE key and stores it along with encrypted medical files using type based proxy re-encryption technique. In this solution data owner only needs one key pair to protect his medical data and can choose the proxy re-encryption for each category of her PHR data according to his trust and privacy concerns.

Using lazy encryption, the affected encrypted file and user secret keys may only be updated when a user log on to system next time. These characteristics led to usage of lazy re-encryption [13] process which allows cloud servers to update user secret keys and data files .After user revocation process, cloud servers just record information submitted by data owner .When an access request comes from a user, cloud servers re-encrypt the required files and update the requesting user's secret key. This saves a lot of computational overhead since cloud servers are able to associate multiple update/re-encryption operations into one if there is no access request occurring across multiple successive user revocation.

4.8 File Download

User request a file stored in cloud and in response cloud server replies with encrypted file and data encryption key. Authorized user having SK,can decrypt DEK and gain access to shared contents by decrypting file with DEK.The main procedure for file access is as follows:

On receiving the request cloud server first checks whether requested user is a valid system user in UL.If true, they update this user's secret key components to the latest version and re-encrypt the DESs of requested data files using the latest version of PK. Cloud servers will not perform update/re-encryption if secret key components/data files are already of latest version. Finally cloud servers send updated secret components as well as encrypted data files to users.

On receiving this user first verifies if the claimed version of each attribute is really newer than the current version he knows. For this purpose, he needs to verify the data owner's signature on the attribute information and corresponding public keys. If correct, the user further verifies each secret key component returned by cloud servers is correctly computed. Finally he decrypts the DEK with secret key and then decrypts the data file using DEKs.

4.9 Break Glass Access

Break glass access is a technique for breaking a patient's access control in emergency conditions. During emergency situations like unconsciousness, accidents etc, the medical staffs need to have access medical data without patient's consent. The medical data is in encrypted form in cloud and so they will need some temporary authorization to decrypt data. Here we apply the break glass method implemented using attribute based encryption [15].Along with access control policies in ABE, a set of emergency attributes is also defined. These emergency attributes can be used to access the encrypted medical data. When the medical staff requests emergency attributes from emergency department, current emergency level is checked and a copy of emergency attributes is released which is used to decrypt the medical

data. Then the copies of attributes are deleted. For supporting the activation and deactivation of emergency attributes by CSP, main approach is that decryption is only possible if the required emergency attribute is active during both encryption and decryption. After patient become well, he can get back access control by redefining access control policy and emergency attributes.

5. PERFORMANCE ANALYSIS

This section evaluates the performance of the main method in terms of computational complexity produced by each operation.

5.1 Data Preparation

This is the preliminary step for exchanging data files among secured and unsecured domains. This is mainly done by the data owner. The owner has to select symmetric encryption key DEK from key space. The main computational overhead is directly proportional to the size of data file, i.e. O(S).

5.2 New File Creation

Here the main function is the encryption of the file using symmetric key DEK and encryption of DEK using KP-ABE. The computation overhead of this process is directly proportional to the number of attributes A used for creating access policy, i.e. O (A).

5.3 New User Registration

This process mainly involves data owner, cloud server and the user. The main computational overhead is due to generation of user secret key and encrypting this secret key with user's public key. So the complexity is equal to O (L), where L is the number of leaf attributes in the user's access structure.

5.4 User Revocation

This operation mainly involves data owner, user and cloud servers. This process involves the updating of access structure of remaining users for preventing access of revoked user. Considering n' users are revoked, the overhead would be O(n') where n'<n.

5.5 File Download

For downloading data files, user has to get DEK and encrypted file from cloud server. The DEK is decrypted using secret key and this DEK is used to decrypt the cipher text. Since DEK is encrypted with KP-ABE, the decryption complexity is equal to access structure P used for creating secret key, i.e., O (P).

6. SECURITY ANALYSIS

In this section, we examine the security aspects of the proposed scheme.

6.1 Data Confidentiality

We analyze data confidentiality by comparing it with symmetric encryption algorithms like DES, triple DES, and AES.The main challenge is to combine cloud data security with owner's security policies. The cloud server is not able to know the content of encrypted data and key is concealed using KP-ABE. By using double encryption, the key size is increased and so brute force attack becomes difficult on encrypted text.

6.2 Combined Fine and Coarse grained Read Access Control

Here, the patient is able to define access control policy for each user. The access policy of each user can be defined as a unique logical expression over these attributes to reflect the scope of data files. Here both user centric and server centric access control is used to achieve more security of data. The user encrypted data is again encrypted by cloud server. The corresponding key is distributed to the authorized user.

6.3 Authentication and Integrity

The communications between data owner and cloud service provider is authenticated by encrypting data keys and DEKs.At the time of registering new user ,user is authenticated at owner by signing with his private key. Integrity is achieved by digital signature algorithm. The encrypted file is signed with owner's private key and cloud server can verify the signature by owner's public key. It proves the identity of sender.

7. RESULTS AND EVALUATION

In this section, we show the performance assessment of the proposed scheme.

7.1 Performance and Evaluation Setup

To measure the proposed scheme's performance for owner and user, evaluation process is carried out on 32 bit Intel Pentium laptop, Windows 7 with 2 GHz Dual Core Processor and 2 GB RAM. We evaluate the proposed scheme on three different encryption algorithms,DES,Triple DES and AES.To evaluate its performance on a cloud server we choose Microsoft Azure [11] as a cloud service provider. Cloud server is only responsible for data storage and key management.

7.2 Computation Overhead

The main step in the proposed scheme is defining the access policy P under which DEK is hidden.KP-ABE implementation supports two kinds of policy generation mainly due to the types of attributes associated with SK.First type is simple literal attributes with leaf nodes of P, e.g., Physician, Nurse, Pharmacist, Hospital A. The second type is complex attributes with logical connectives, e.g., (profession=nurse)^(duty=ICU)^(institution=hospital C).Fig 3 illustrates the generation of secret key with 10 different simple attributes would take 0.1 seconds. But it would take at most 18 seconds to generate a secret key associated with complex attributes.

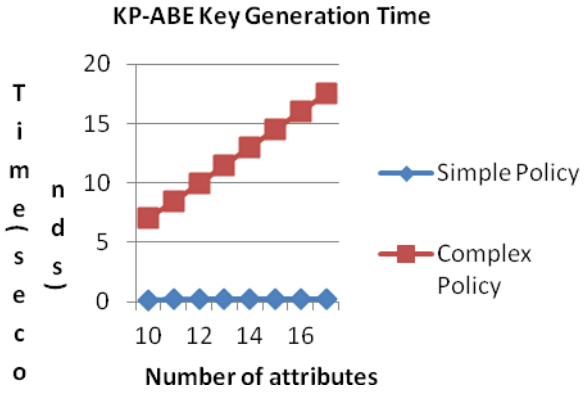

Figure 3: Computation overhead of secret key generation process of KP-ABE.

User executes KP-ABE encryption process to encrypt DEK.Decryption is primarily needed when user get the DEK for first time. As KP-BE support two types of policies, fig 4(a) and 4(b) shows the computation overhead of KP-ABE decryption process over 56 bit,128 bit and 256 bit keys of

DES,Triple DES and AES respectively.KP-ABE exhibits same decryption time for different sizes of cipher text, ,encrypted under similar policies. However type of attribute does not affect the complexity of decryption process.

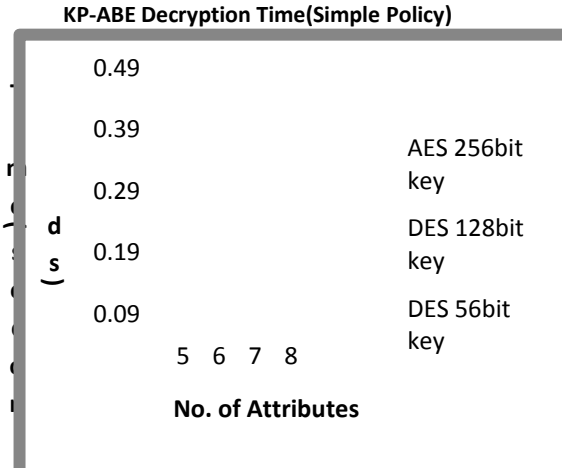

Fig 4(a): Simple Access policy Computation overhead

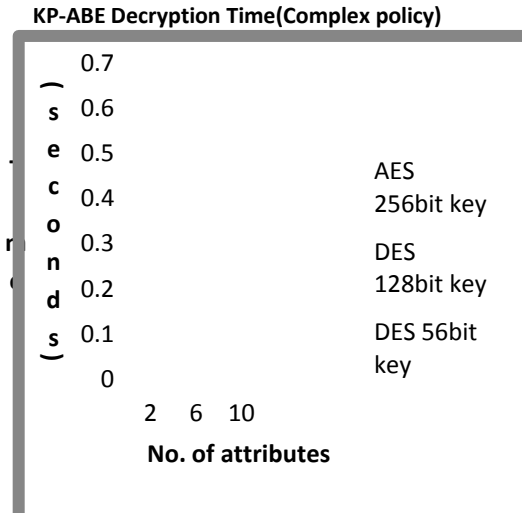

Fig 4(b): Complex Access Policy Computation Overhead

7.3 Communication Overhead

We can compute the amount of data exchange between cloud server and its users. The decryption keys streaming between entities can greatly affect the invoice which owner got from cloud service providers. In this context, for communication overhead, we measure the amount of encrypted data flowing between the entities except from encrypted file; there is no restriction on symmetric algorithm.

In addition to encrypted file,DEK is stored on the cloud server.KP-ABE can greatly reduce the amount of data exchanged between entities in order to achieve fine grained access control, if policies are defined for groups instead of individual users. Figure 5 shows that size of secret keys is in direct proportion to the number of attributes associated with it. For complex policy KP-ABE outputs bigger sized keys as compared to simple policy.KP-ABE is applied to different size of symmetric encryption algorithm decryption keys. The variation in size of encrypted file is due to presence of logical operators in the complex attributes policy.

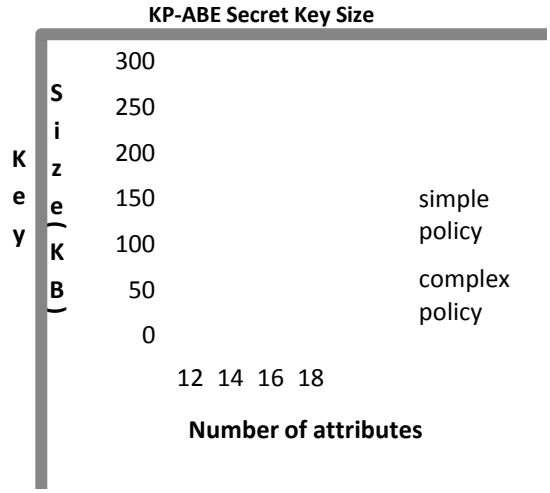

Fig 5: KP-ABE user secret key size with different attributes

8. CONCLUSION AND FUTURE SCOPE

Cloud storage is currently a feasible industrial solution for medical records sharing and cloud computing is envisaged as a short term solution to reduce the computing costs in biological groups. The main aim is to achieve fine-grained data access control in cloud based medical records sharing. To achieve this goal, exploiting KPABE and uniquely combining it with techniques of proxy re-encryption and lazy re-encryption are performed. Moreover, this scheme can enable the data owner to delegate most of computation overhead to powerful cloud servers. Thus, the proposed scheme enhances data privacy and confidentiality in the medical data sharing system against third party providers as well as adversarial outsiders without disclosing contents. As future scope we are presenting the problem of how encryption will affect dynamic data operations such as query, insertion, modification, and deletion. Data anonymization and privacy preserving techniques will increasingly assume greater importance in cloud based medical data exchange and more mainstream research is required in this area.

9. REFERENCES

[1] Zhuo-Rong Li1, En-Chi Chang1, Kuo-Hsuan Huang1, Feipei Lai2*,"* A Secure Electronic Medical Record Sharing Mechanism in the Cloud Computing Platform**,"** IEEE 15th International Symposium on Consumer Electronics 2011, pages 450-457.

[2] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. Of CCS'06*, 2006.

[3] 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 *Proc. of VLDB'07*, 2007.

[4] R. Zhang and L. Liu, "Security Models and Requirements for Healthcare Application Clouds," Proceedings of IEEE 3rd International Conference on Cloud Computing, 2010, pages 268-275.

[5] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Scalable secure file sharing on untrusted storage," in *Proc. of FAST'03*, 2003.

[6] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proc. of ESORICS '09*, 2009.

[7] M. Atallah, K. Frikken, and M. Blanton, "Dynamic and efficient key management for access hierarchies," in *Proc. of CCS'05*, 2005.

[8] Mrinmoy Barua, Xiaohui Liang, Rongxing Lu, and Xuemin (Sherman) Shen "PEACE: An Efficient and Secure Patient-centric Access Control Scheme for eHealth Care System," The First International Workshop on Security in Computers, Networking and Communications,2010.

[9] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," in *Proc. of NDSS'05*, 2005.

[10] Benaloh, J., Chase, M., Horvitz, E., and Lauter, K. (2009) Patient controlled encryption:ensuring privacy of electronic medical records. Proceedings of the 2009 ACM workshop on Cloud computing security, New York, NY, USA, pp. 103{114, CCSW '09, ACM.

[11] Microsoftwindows Azure. Http://www.microsoft.com/azure/.

[12] Ran Canetti and Susan Hohenberger. Chosen-ciphertext secure proxy re-encryption. Cryptology,ePrint Report 2007/171, 2007.

[13] M. Backes, C. Cachin, and A. Oprea, "Secure key-updating for lazy 0revocation," Technical Report RZ 3627, IBM Research, Aug. 2005.

[14] Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres, "A Break in the Clouds: Towards a Cloud Definition," *in ACM SIGCOMM Computer Communication Review*, Volume 39, Number 1, January 2009

[15] Shucheng Yu,Cong Wang,Kui Ren and wenjing Lou,"Acheiving Secure,Scalable,and fine grained Data Access Control in Cloud Computing",IEEE INFOCOM 2010.

[16] Luan Ibraimi,QiangTang,Pieter Hartel,and William Jonker,"A Type and Identity based Proxy Re-encryption scheme and its application in Healthcare",SDM '08 Proceedings of the 5th VLDB workshop on Secure Data Management,2010.

[17] Achim D.Brucker,helmut Petritsch and Stefan G.Weber,"attribute based encryption with break glass",Springer 2010.

[18] Saman Iftikhar, Wajahat Ali Khan, Maqbool Hussain, Muhammad Afzal, Farooq Ahmad," Design of Semantic Electronic Medical Record (SEMR) system as SaaS service model for Efficient Healthcare",IEEE 3rd International conference on cloud computing 2010,pages 344-347.