The Privacy Based Secure Sharing Of Personal Health Records a Method in the Cloud

[1]M/s Aaliya Shaikh, [2]Mr. Rathod V .U
[1]Students, Department of Computer Engineering,
[2]Prof., Department of Computer Engineering,
Vishwabharati Academy's College of Engineering, Ahmednagar (MH), India

Abstract : Now a days every information is stored and shared on the cloud. And so the medical record especially Personal Health Record (PHR) is an important part of health information exchange, that is need to be stored at cloud servers. But there are various privacy problems as personal health information could be discovered to unauthorized people. That need guarantee of the patient control over to their own PHRs, in this method encryption of the PHRs is done before the storage on cloud. But still issues like risks of privacy, efficiency in key administration, flexible access and efficient user administration, have still remained the important challenge toward achieving better, cryptographically imposed data access control. In this research development, we develop a mechanism for control of data access to PHRs stored in cloud servers. To achieve this efficient and modular data access control for PHRs, we provide El-Gamal encryption approach for the encryption to each PHR file. For this system method already tried to focus on the multiple data ownership scheme also dividing the users into security domains that highly reduce the key management complication for owners and users. Here the system takes patient privacy as serious issue and guaranteed it by exploiting multi-authority Encryption. Our main aim is not only privacy but also system's scheme try to enable modification of access policies or file attributes, and break-glass access under emergency situations. Our proposed scheme shows Extensive analysis and experimental results are presented for security and efficiency of PHR.

Keywords— El-Gamal encryption; PHR; access policies; medical record ;cloud computing; privacy;

I. INTRODUCTION

As Cloud computing means on demand access and storage of data and programs over the internet instead of using computer's hardware and software. Data security in cloud computing is major problem. So to safe guard security, different attribute based encryption schemes are used for encryption before outsourcing data to cloud server. Personal Health Record (PHR) service is an private data from patients perspective. For patients health information need to be stored and exchange in confidential manner. So this paper presents model that also secure storage, analysis and transfer of the information. It allows patients to create, update and manage personal and medical information. Also they can provide health care providers their data's control and share their medical information with other users as well . Advance technology of cloud computing where PHR has gone through substantial changes. Most health care providers and different vendors related to healthcare information technology started to treat PHR services as a simple storage service. That turns them into complicated social networks like ervice to patient for sharing health information to others with the help of cloud computing. PHR data is hosted by the third party cloud service providers in order to enhance its interoperability.

However, there have been serious issues in retribution these data to the cloud server with regards to security. So we encrypt the PHRs before sending to cloud.

Because of that many issues like risks of exposure privacy, scalable key management, flexibility in access and efficient user revocation, have remained the most important challenge toward achieving fine-grained, cryptographically enforced data access control. For achieving a incompressible data and expandable data control access on client's data, the original patient-centric framework is used.

Here this methodology that takes PHR access control mechanism managed by patients themselves and achieves privacy. That preserves the confidentiality of the PHRs by restricting the access of the unauthorized users. Generally, there are two types of PHR users in the proposed approach, namely: (a) doctor who shares medical record of patient, nurses and family members of patient (b) Patient itself or owner. The owners of the PHRs are permitted to encrypt first and then upload file on cloud by selectively granting the access to users over different portions record of the PHRs. Each member is granted access according to its authority related to patient. That PHR owner set authority to a certain level depending upon the role of the user. The levels of access granted are defined in the Access Control List (ACL) according to various categories of users by the PHR owner.

So here, the family members or friends of the patients may be given full access over the PHRs by the owner. In case, the insurance company representatives may only be able to access the portions of PHRs containing information about the health insurance claims while the other confidential medical information, such as medical history of the patient is restricted for such users.

II. LITERATURE SURVEY

A Public key cryptosystem Encryption

The PKE technique requires two separate keys; one of the keys is private whereas the other is public. Solutions based on the PKE are secure but using the PKE alone seems computationally less efficient due to the slower operations and the larger key sizes.

Public key cryptosystem that can aggregate any set of secret keys into a single compact aggregate key that makes power of all the keys being aggregated. But it did not focus on how it can help patients to have fine grained access control and revocation of access control. Also how it will achieve confidentiality, authentication and integrity of their PHRs at the same time [1].

B Symmetric key cryptosystem

The SKE uses the same keys for encryption and decryption. The SKE based algorithm currently in use and acting as standard is the Advanced Encryption Standard (AES). The AES was recommended as a standard by the National Institute of Standards and Technology.

This scheme enables patient-centric access control over PHR data. The proposed scheme ensures patient-centric fine-grained access control, also revocation of access control using symmetric key cryptosystem and proxy re -encryption (PRE) scheme. But drawback of this scheme is, each file category is encrypted with distinct secret key. The patients have to provide the corresponding secret keys whenever a data user (e.g. Doctor or nurse) wants to update PHR. Besides this, the scheme is based on proxy re-encryption scheme that needs data owners to have too much trust on the proxy that it only converts cipher texts according to his instruction [4]. A PRE scheme allows data owners to empower the proxy for converting the cipher texts encrypted under his public key into ones for data users. Hence it is desired that storage server should not have proxy resided in it. That results in increase communication overhead since every decryption requires separate interaction with the proxy So in this paper, we redesign the scheme that ensures properties: (1) confidentiality (2) integrity (3) authenticity (4) patient -centric fine-grained access control, and (5) revocation of access control.

C Attribute-Based Encryption (ABE)

Attribute-Based Encryption (ABE) is a cryptographic primitive based on the PKE where the messages can be encrypted and decrypted on the basis of user attributes. A cipher text can be decrypted only when the attributes and the decryption keys are available [4]. The data encrypted by a single owner is subsequently shared with multiple users by distributing the keys. To enable data owner to delegate the computational tasks to the untrusted cloud servers, the access policies based on the attributes are enforced.
The approach ensures the accountability of the users' secret keys. In the proposed approach the tasks of re-encrypting the data files and updating of the secrete keys are delegated to the cloud servers. To deal with the heavy computation overheads caused by re-encryption of data files and update of secret key, the KP-ABE, PRE, and lazy re-encryption are combined. In paper proposed an EHR solution, that depends on smart cards and RSA which enables patients to store their medical records on hybrid clouds. In this approach, the hospital's private cloud and the public cloud, in this way patients' medical records are stored in two types of cloud. At first the medical records being accessed by the owner of data, i.e., the doctor who created the records. They can directly access the records from their public cloud or from the private cloud [7]. The second case is that of the medical records being accessed by other hospitals, they must first seek permission from the data owner also here authentication and authorization is important .That also provide a solution for emergency situations. However, doctors have access control for the medical records and their computing load is heavy is bottleneck.

D Conditional Proxy Re-encryption (C-PRE)

Proxy-re encryption (PRE) –is a cryptographic basis that makes semi-trusted proxy to convert the cipher text encrypted under the public key of one user into a cipher text that can be decrypted through the other user's private key.
A solution that allows patients to specifically make a policy for supporting a acute access control. They make use of Conditional Proxy Re-Encryption for enforcing sticky policies and providing users with write privileges of PHRs. The users they sign the modified PHRs whenever they finished writing data to their PHRs. However, it is difficult to correctly verify who signed the PHRs users as the sign of PHRs using the signature key of the PHR owner [3].

III. PROPOSED SYSTEM

A. Admin This module provides secure patient-centric PHR access and efficient key management at the same time. According to the different users' data access requirements main idea is to divide the system into multiple security domains (namely, public domains (PUDs) and personal domains (PSDs)).

B. Server A semi-trusted platform where the system ensures that each party in our system is preloaded with a public/private key pair, and entity authentication can be done by traditional challenge-response protocols. In the framework, there are multiple owners, multiple AAs, multiple SDs and multiple users.

C. Security of data The owners upload El-Gamal-encrypted PHR files to the server. For encryption of PHR El-Gamal algorithm technique is used for key generation technique is used for Encryption.

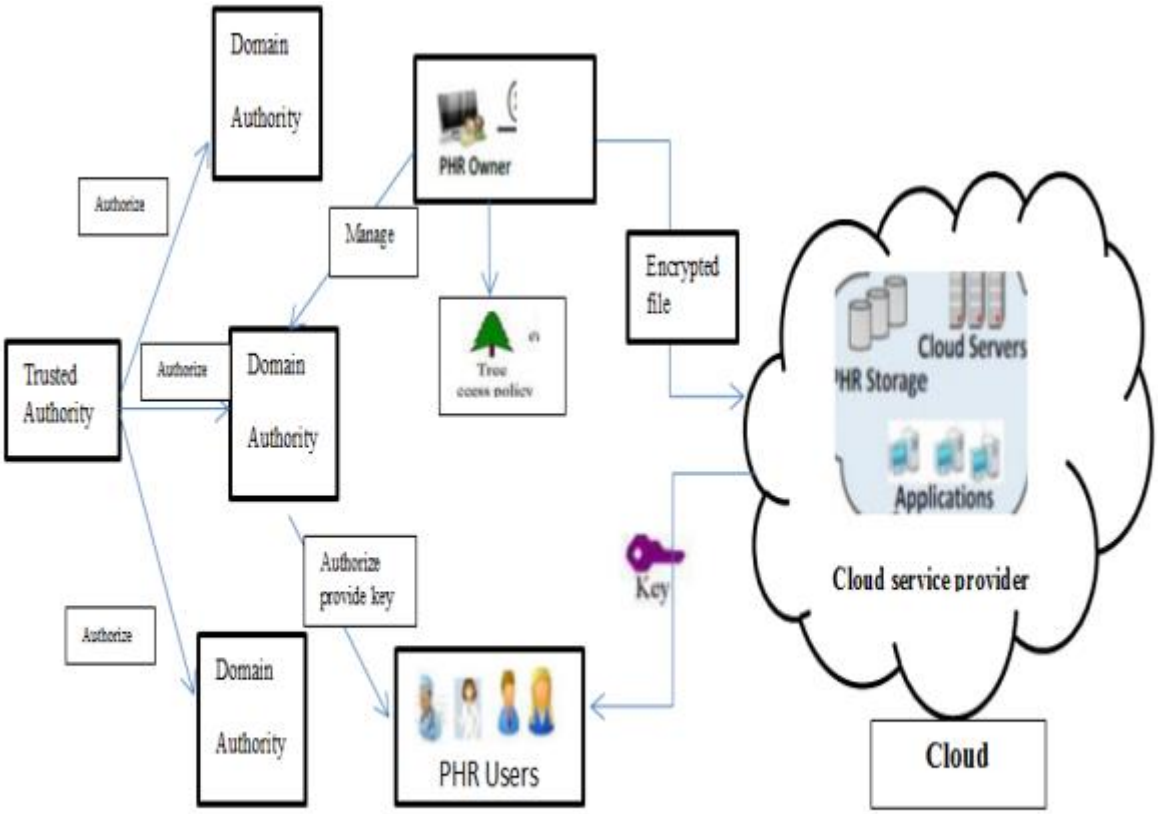


Figure 1: System architecture

D. Software Requirements Specifications

1) Hardware Requirements
- Pentium IV Processor and above
- Minimum RAM 512 MB
- Minimum 40 GB Hard Disk

2) Software Requirements
- OS Requirements: Windows 7 onwards
- NetBeans 8.0.1
- JAVA JDK 1.7 and above
- MySQL Server 5.5

IV. ALGORITHM

A. El Gamal Encryption

Here we are using the different secret key k to encrypt as well as decrypt data. This will use to convert the plain text to cipher text and again cipher text to plain text. Here we have used three basic functions,
- KeyGenSE: k is the key generation algorithm that generates κ using security parameter 1.
- EncSE (k, M): C is the El-Gamal encryption algorithm that takes the secret κ and message M and then outputs the cipher text C;
- DecSE (k, C): M is the El-Gamal decryption algorithm that takes the secret κ and cipher text C and then outputs the original message M.

B. Encryption Algorithm

1) In this case four r1,r2,r3,r4 $\in$ Zq$*$ random variables are generated.
2) The variable r*i* is of the PHR to encrypt i-thpartition. Each separately by the client application is partition is encrypted.
3) The application to perform encryption/decryption on logical partitions of the PHR.
4) The encryption done on partitions of the PHR is performed as follows.

Cper=Z^{r1}PHRper
Where to the personal partition PHRper referred of the PHR and Cper is the semi-encrypted file that contains the personal partition as encrypted text.
5) $\mathcal{C}$ins is the semi-encrypted file that contains the insurance partition as encrypted text in addition to the $\mathcal{C}$per that was encrypted in the previous step And PHRins refers only to the insurance partition of the PHR.
6) $\mathcal{C}i$ns=Z^{r3}.PHRins
Where $\mathcal{C}$med=Z^{r3}.PHRmed
Cmed is the semi-encrypted file that contains the insurance partition as encrypted text in addition to the Cper and Cins that were encrypted in the previous steps Where PHRmed refers only to the medical information partition of the PHR.
7) C=Z^{r4}.PHRpres
Here, $\mathcal{C}$ represents the complete encrypted file that contains all the partitions in the encrypted form. Where PHRpres refer prescription information partition.
8) Parameters are

$$Rper\_P = gr1xp$$
$$Rins\_P = gr2xp$$
$$Rmed\_P = gr3xp$$
$$Rpres\_P = gr4xp$$

xp is the private key The parameter R is used to produce the re-encryption key. The parameters Rper_P, Rins_P, Rmed_P, and Rpres_P are transmitted along with the file identification for which these parameters are generated.

C. Decryption Algorithm

1) The user downloads the $\mathcal{C}$ directly from the cloud
2) Afterwards the user U requests for corresponding R parameters that are used for decryption.
3) Checks the ACL for the re-questing user and determines whether the access to the partition for which the user has requested R, is granted by the PHR owner or not.
4) According to the access per-missions specified in the ACL, the corresponding parameters and will send those to the re-questing user
5) Then calculates the encryption key and R and transmits it to the user U by $RKP \rightarrow U = gxUxP$ (14)
Where $RKP \rightarrow U$ is the re-encryption key from patient$P$ to user U, xU and xP are the private keys of U and$P$, respectively.
6) Subsequently, the parameters R for all of the partitions corresponding to the user U are calculated according to the following equations.

$$RperU = e(RKP \rightarrow U, RperP) = e(gxUxP, gr1xP) = e(g, g)r1xU = Zr1\ xU$$

Where Rper_U is the parameter used to decrypt the partition 'personal information' and is applicable for the user U.
7) Similarly, R parameters for other partitions corresponding to user U are calculated in Encryption algorithm

$$Rins\_U = e(RKP \rightarrow U, Rins\_P) = e(gxUxP, gr2xP) = e(g, g)r2xU = Zr2\ xU$$

$$Rmed\_U = e(RKP \rightarrow U, Rmaed\_P) = e(gxUxP, gr3xP) = e(g, g)r3xU = Zr3\ xU$$

$$Rpres\_U = e(RKP \rightarrow U, Rpres\_P) = e(gxUxP, gr4xP) = e(g, g)r4xU = Zr4\ xU$$

8) The above given parameters are provided to the user U that decrypts each of the partitions based on the following equations.

$$PHRper = CperRper\_U1xU$$
$$PHRins = CinsRins\_U1xU$$
$$PHRmed = CmedRmed\_U1xU$$

9) $$PHRpres = CpresRpres\_U1xU$$
10) The decryption of the last partition will result in complete PHR in plain form.

D Secure Share Algorithm

1) Make registration using parameters
2) Login information is authenticated.
3) Choose PHR file to upload
4) Does Encryption and upload file
5) Also access policy set by user Label of each partition, for instance personal information, medical information, insurance information, and prescription information
6) Role that has access to any particular partition (any role may be given access to more than one partitions), like doctors may be given access to medical in-formation
7) Initial members of family/friends to give access
8) Default access (if any) in case of new member

9) Upload the file on server
10) Generated keys are shared to patient based on policy

V. MATHEMATICAL MODEL

- PHR Owner
  S1= {do, d1, F1, F2, F3, d2, d3, d4, e2}
  d0: Registration & Login f1: Encryption of PHR f2: Decryption
  f3: Secret key generation d1: upload PHR file
  d2: store PHR file and attribute d3: transmit parameters
  d4: set control on file access

- Staff(PHR User): S2 :{ e1, e2, e3, F2}
  e1: PHR file access request e2: PHR file download
  e3: Request key to decrypt F2: Decryption ()

- Setup or Server: S3={F1,F3,t1,t2,t3,d3,e3,d3}
  T1: Re-Encryption calling F1function T2: Update & mange keys
  T3: Distribute key

- Cloud: Set (C)={d1,d2,d4,e1,e2,e3,c0,c1} C0-store encrypt file (Storage of encrypting file using El – Gamal Algorithm)
  C1-send encrypted file to data user

- Function Encryption () Input: Attribute Value (Attr). Get Byte [] (B1) of that Attr.
  Generate Public Key (Pk). Perform Encryption on B1.
  Convert B1 into string (EAttr).

- Function Decryption ()
  Input: Encrypted attribute value (EAttr) Convert EAttr into byte [] (B2).
  Generate Private Key. Perform Decryption on B2.
  Convert B2 into string (DAttr).

- Secret Key ()
  Input: Private Key (see Decryption) and No. of Authority (NAuth) =10.
  Get Length of private key: Length = PrivateKey.Length. To become private key multiple of NAuth (i.e. 10) pad it by zero (0).

S1UnoinS2={do,d1,F1,F2,F3,d2,d3,d4,e1,e2 ,e3}
S2UnoinS3={do,d1,F1,F2,F3,d2,d3,d4,e1,e2,e3, t1,t2,t3}
S1UnionS3={ do,d1,F1, F2,F3,d2,d3,d4,e2,e3 ,t1,t2,t3}
C1= S1intersectionS2= {d1, e2, F2}
C2= S1intersection S3= {F1, F3, d0, d3, d4}, C3 = S2intersectionS3={e3}
S1intersection C= {do, d1, d2, d3, d4, e2}, S2 intersection C= {d1, e1, e2, e3}, S3 intersection C= {t1, t2, t3, d0, d3, e3, d4}

C1 intersection C= {d1, e2} C2 intersection C= {d0, d3, d4} C3 intersection C= {e3}

- **Success Condition**: Success system when the PHR file will access with authorized user without any problem.
- **Failure Condition**: Failure system when the server will fail during the download the file.

VI. RESULT AND DISCUSSIONS

We propose a sensitive policy; privacy based approach to the PHR files sharing.

- For minimizing the loss of the uploaded files from unauthorized patient/user.
- For minimizing the disclosure risk.
- To maintain the diversity among the uploaded PHR files.
- Minimized security on files on sharing sites.

The performance of this method based on the access policy set to each PHR file which is shown in Table 1 for different access policy result is successful. The Encryption /Decryption time, Key generation and total time of process should be minimum for good performance.

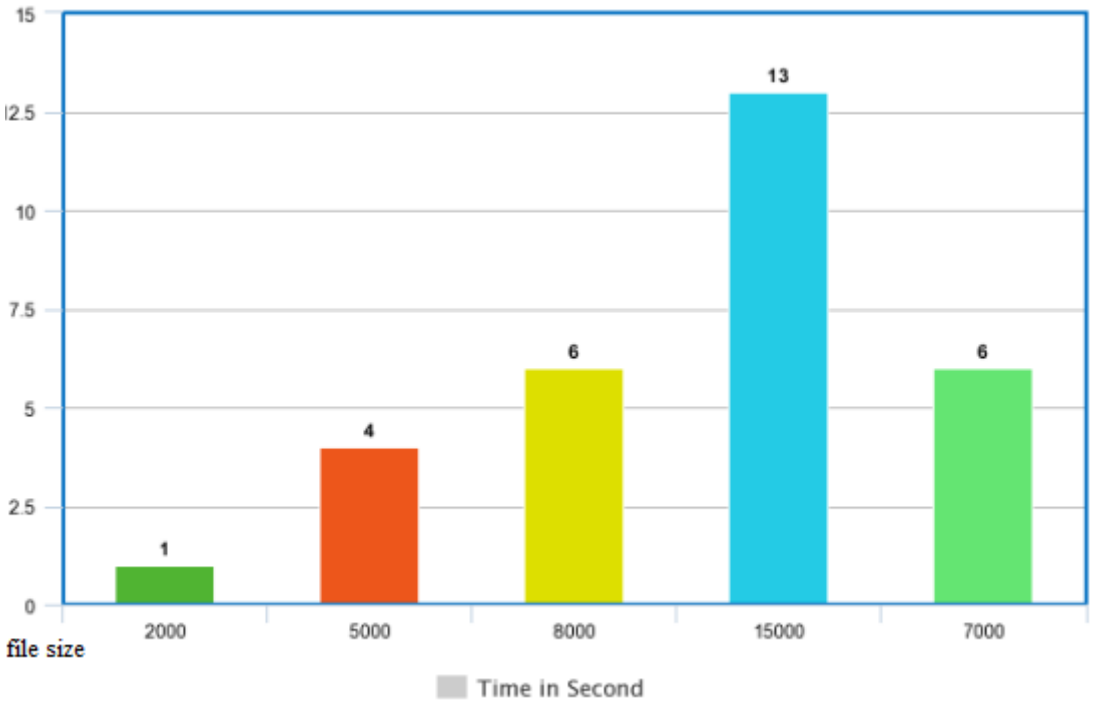
TABLE 1: SHOWS THAT ACCESS POLICY SET TO THE PHR FILE UPLOADED

Sr.no	Patient Name	File name	Access Policy	Result
1	Jhon	Neurotherapy_Treatment.doc	Doctor/ Patient	Successfully Apply Policy
2	Peter	Heart_Treatment.doc	Doctor/Patient	Successfully Apply Policy
3	Ganesh	Daily_Treatment_Plan.doc	Receptionist	Successfully Apply Policy
4	Linda	Tablets_Plan.doc	Receptionist	Successfully Apply Policy

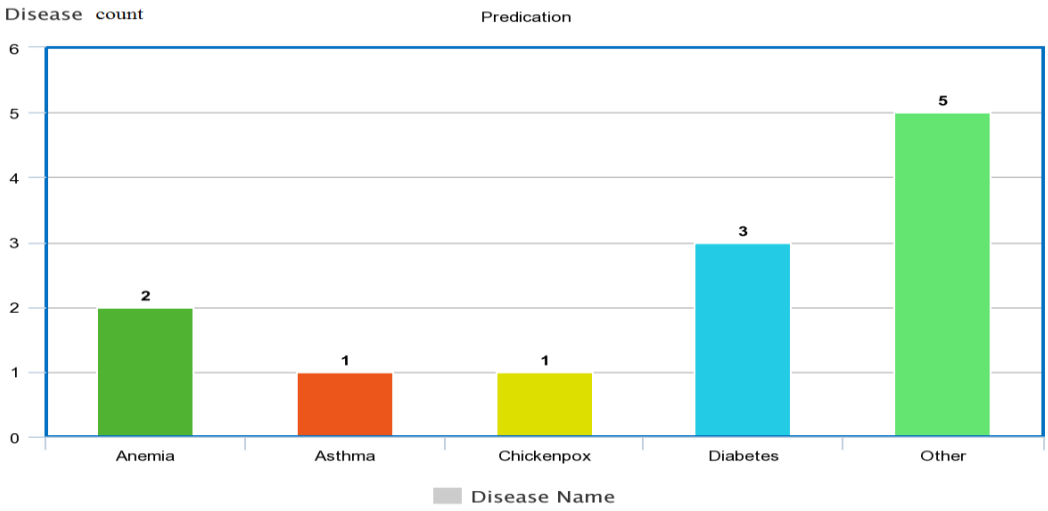
The result is to meet expected output on given experimental data file. The system Privacy Based Secure Sharing Of Personal Health Records a Method in the Cloud that needs to upload patients information in encrypted form on cloud with policy set on cloud and that file is decrypted with key based on authentication also policy based access. The expected result is to generate key, encryption of file with secure storage on cloud to avoid misuse of information some accessibility rights are set and key is provided based on that to guarantee confidentiality. Also valid system user cannot obtain the re-encryption parameters for a PHR partition for which access is not granted to the user. The server gives user access permission and keys to user. The Actual output works with patient registration and login also patient upload PHR file and set policy send on doctor desk for checking.

The PHR file is encrypted before uploading. Decryption done with key asked from user firstly policy is checked the key provided for decryption of data.

Graph 1: ENCRYPTION PROCESS TIME IN EXISTING SYSTEM AND PROPOSED SYSTEM DIFFERENT FILE SIZE



The Graph 1 shows better performance of proposed system as compared to exiting with minimum time in encryption for different file size. And so the turnaround time will also be less. Sample file is tested with 1kb, 4kb, 6kb, 13kb, 6kb that required 2sec, 5sec, 8sec, 15sec, 7sec in proposed system.



Graph 2. Indicate that which disease is happen to how may patient.

VII. CONCLUSIONS

The increase of storage on cloud and work on distributed system that need to ensure healthcare systems based on cloud storage should be privacy based, how to protect PHRs stored in the cloud is a central question. Also it enforces a patient-centric access control to different portions of the PHRs based that to be easily accessible. This implemented system is a Cryptographic technique that are getting more versatile and often involve multiple keys for a single application which increases the key management overhead but efficient in sense of privacy. In this proposed method, also discuss how the confidentiality, and authentication of PHRs can be achieved This system also enables a patient to exercise complete control over their PHRs and perform revocation of access rights. Furthermore we can do project work on real time application using the hospital record based on global server that is to be easily accessible and potential to patient and Doctors.

VIII. ACKNOWLEDGMENT

I wish to express my profound thanks to all who helped us directly or indirectly in making this paper. Finally I wish to thank to all our friends and well-wishers who supported us in completing this paper successfully I am especially grateful to our guide for him time to time, very much needed, valuable guidance. Without the full support and cheerful encouragement of my guide, the paper would not have been completed on time.

REFERENCES

[1] W. Tzeng, S. Chow and C. Chu “Key-Aggregate Cryptosystem for Scalable Data Sharing in Storage”, IEEE Transactions on Parallel and Distributed Systems, 25 (2): 468- 477, 2014.
[2] Yao Zheng, Li and Yu, “Scalable and Secure Sharing of Personal Health Records in Cloud Computing Attribute-Based Encryption”, IEEE Transactions on Parallel Distributed Systems, 24(1), pp. 131-143, 2013.
[3] Yu, H., Leng, C., Wang, J., & Huang, J. “Securing Personal Health Records in the Cloud by Sticky Policies,” TELKOMNIKA Indonesian Journal of Electrical Engineering, 11 (4), 2200-2208, 2013.
[4] S. Yu, C. Wang, K. Ren, and W. Lou, “Achieving secure, scalable and fine-grained access in cloud” IEEE INFOCOM Proceedings, pp. 1-9, March 2010
[5] Y. Huang, En Chang, and Shao Wang, “A Patient-Centric Access Control Personal Health Records Cloud”, Fourth International Conference on Networking and Distributed Computing, 2014. [6] G. N., Dixit “Patient Centric Frame Work For Data Access Control In Cloud Server”, International Journal of Engineering, 2 (4), 2013.
[7] Chen, Y. Y., Lu, J. C., & Jan, J. K. “A secure EHR system based on hybrid clouds,” Journal of medical systems, 36 (5), 3375 - 3384, 2012.
[8] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, “Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records”, Proc. ACM Workshop Cloud Computing Security (CCSW 09), pp. 103 -114, 2009.
[9] Chen L., Chen D., Fan X. “Securing Patient-Centric Personal Health Records Sharing in Cloud Computing”, China Communications, Supplement No.1, 2014.
[10] O. Pandey, V. Goyal, B. Waters, “Attribute-Based Encryption for Fine-Grained Access Control over Data”, Proc. 13th ACM Conf. Computer and Comm. Security (CCS 06), pp. 89-98, 2006
[11] M. Chow, M. Chase “Improving Privacy and Security in Multi-Authority Encryption”, Proc. ACM Conf. Computer and Comm. Security, pp. 121-130. 2009.
[12] S. Hohenberger, R. Canetti, “Chosen-Ciphertext Secure Proxy Re- Encryption”, Proc. 14th ACM Conf. Computer and Comm. Security (CCS 07), pp. 185-194, 2007.