International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online)), Volume 2, Special Issue 3, February 2014)

International Conference on Trends in Mechanical, Aeronautical, Computer, Civil, Electrical and Electronics Engineering (ICMACE14)

Secure Alternate Viable Technique of Securely Sharing The Personal Health Records in Cloud

K.S. Aswathy[1], G. Venifa Mini[2]

[1]M.E. Student, [2]Assistant Professor, Computer Science and Engineering, Noorul Islam University, Kumaracoil, India

[1]ash.cse18@gmail.com, [2]venifamini@yahoo.co.in

***Abstract*—In Cloud environment, variety of resources can be accessed freely or by the pay-per-use schemes. In addition to this kind of resource sharing, it also provides different types of services and sharing schemes which are very effective in the cloud applications. One among them is the personal health record sharing. It is a patient centric framework which exchanges the health details of the patients to the required users in a secure and safe manner. Different encryption schemes are used to securely share the data but all suffers from certain security issues. Hence we propose a new technique called Secure Alternate viable technique which overcomes all the security issues in sharing the personal health data.**

***Keywords*—Key Generation Module (KGM), Transitional Encapsulation Module (TEM), Alternate Decryption Module (ADM), Additive Homomorphic Encryption (AHE)**

I. Introduction

Cloud computing technique is everywhere. Nowadays people often prefer to make use of cloud environment for data sharing and accessing very effectively. In cloud, anyone can freely access any data for their use. It also has a pay-per-use service in which people instead of buying the whole product can simple pay only for their use. Time and cost are saved effectively due to this technique. To share and access data, there is no need to know the owner details of the data. It is very simple and effective.

In the current world, people suffer from different health issues. They often go to different hospitals to get consulted and get their treatments. However they also suffer from health issues they do not wish to share with anyone. Hence they are in need of a new technique to know about their health conditions. Cloud environment provides one such service. Personal health record sharing in cloud is an efficient patient centric framework of health information sharing to authorized and required users. The patients itself will have the central control over their own data being uploaded in cloud.

Different types of users (doctors, family, friends etc) will be available in cloud. Based on the privacy allocated to each user by the data owner, the data are shared and get connected. The main advantage of this technique is that the patients can consult with doctors online can get treatment based on their suggestions.

To secure share these health details, different types of encryption schemes are used. Attribute based Encryption technique is one among them. Other encryptions schemes are also used but all suffers from certain issues such as key size, complexity, time, cost etc.

In this paper, we propose a new technique for encrypting the data which uses the Additive Homomorphic Encryption. In normal, all HE concepts depend on RSA managed methods, such that the encryption is done by using a pair of keys retrieved from a process of very large prime numbers. The security of the system relies on the key size used for generation of the Prime number and the subsequent generation of keys. The complexity of security increases when key size used for encryption is increased. This will need more memory requirements at the receiving end leads to the cost factor at clients end. An alternate technique called Secure Alternate Viable (SAV) Encryption is a technique which uses three modules for securely transmitting the data in cloud. It uses a Key Generation Module (KGM) to generate the key for encryption, a Transitional Encapsulation Module (TDM) to encapsulate the data after performing the Additive Homomorphic Encryption, and an Alternate Decryption Module (ADM) to decrypt the data. The data will be splitted and separately encrypted and stored in cloud. If a consumer wants to consume it, the data will be added by using additive homomorphic encryption and then decrypted and sent to the requested consumer. This is the basic principle in this technique. This is proposed to be a best technique for providing very strong protection to the data in untrusted cloud servers. Hackers will feel very difficult to crack the original data since it is splitted before encryption.

It makes sure that no un-trusted access to the PHR, but allows the authorized data consumers to decrypt the cipher text from the same cloud server using the secret key with Additive operation in the homomorphic encryption. The major benefit of using this technique is that it can use more than one secret key. Normally for one [public key there will be only one private key. However in this technique it uses more than one private keys for a single public key. Hence if the patient wants to modify the secret key he/she do not need to encrypt the data once again.

II. Related Work

The main focus of the paper is to securely share the personal health records in cloud.

Attribute based Encryption technique have been proposed in [1], where the data being stored in cloud are encrypted using the attributes of various users. Only authorized parties can access the data and can encrypt it. However if one can identify the attributes easily, then it is very easy to decrypt the data. In [2], the olden way of sharing health records were done using virtual machines which involves infrastructure as a service concept for providing the service. But it found to be insecure since there was no encryption of data done for providing privacy. They used MyPHR Machines to provide this service but it is very difficult to maintain and not secure. Another technique mentioned in [3] which mainly encrypt the data using homomorphic encryption for providing security from the cloud service provider itself. It involved the use of a semi trusted third party and data packing technique to encrypt and pack the data before outsourcing in cloud. However scalability was the major problem and also it increases the computational cost. Ubiquitous Health Care Services [4] was available based on the concurrent online analysis of the public. In order to achieve this service, a plug-in algorithm framework was used which provides robust, stable and efficient way of satisfying the healthcare services. Commercial portfolio of the software was very expensive and security was not provided. User privacy was another issue and hardware servicing was not possible in this technique. Electronic Health Systems greatly reduces the paper usage and cost.

Further strong technique have arrived [5] which used the fully Homomorphic Encryption technique to evaluate the arbitrary functions directly on encrypted data on untrusted servers. The encrypted data were recrypted again for privacy issues.

Latency occurs due to the computationally intensive multi-million bit modular multiplication. Due to the complexity of this technique, large GPC memory was needed to support efficient implementation. Memory bottle neck was another issue due to usage of large keys. Mobile Health care Services [6] were widely available which provides direct contact between patient-to-physician and patient-to-patient communication via smart phones and tablets however authentication was not provided. Attribute based Authentication was used in which authentication was provided based on the attributes of the individual patients. Since it was difficult to maintain such kind of services, computational cost and communication resources were considered to be the major issue. A federation in cloud [7] was also used for providing secure data outsourcing but high revocation damages the privacy. Fully Homomorphic Encryption technique had been used for single database [8] but there was no technique available for processing the multiple databases in cloud. Cloud-assisted mobile health (mHealth) monitoring [9] was a prevailing mobile communications and cloud computing technologies to provide feedback decision support, to improve the quality of healthcare service while lowering the healthcare cost. However privacy was the major issue since the personal health details were needed to be secured. A new definition of homomorphic signature for identity management in mobile cloud computing [10] have been developed to provide digital signature for authenticating the parties involved in data access but it ought to be found insecure.

From the above considerations, it is well derived that the security of the cipher text is depends on the size of the keys used for encryption. And also it is well known that by increasing the size of the keys, the computation time and other specialized problems related to cost needs to be worried. Increase in demand of cloud services, will leads to the development of many applications or algorithms to effectively manage the cloud services, without facing cost threats. At the same time the system must guarantee the reliable secure communication when compared to the existing works.

Based on the extensive analysis on existing methods, the new technique to be proposed is a major requirement to overcome all these issues. Hence we suggest a new technique based on the study of the issues occurred in the existing systems and we will discuss all these factors in detail.

III. SAV MODEL

A novel patient-centric framework and an alternate mechanism for data access control in cloud are proposed for PHR's stored in semi-trusted cloud servers. The proposed system makes use of Additive Homomorphic Encryption (AHE) techniques to encrypt each patient's PHR file. Different from previous works in secure data outsourcing of data uploaded in cloud, this system focuses on an alternate secure model to puzzle the un-trusted cloud servers or anonymous access controllers while accessing the records. This system uses a Secure Alternate Viable (SAV) algorithm to make sure that no un-trusted access to the PHR, but allows the authorized data consumers to decrypt the cipher text from the same cloud server using the secret key with Additive operation in the homomorphic encryption. SAV uses three modules such as Key Generation Module, Transitional Encapsulation Module and Alternate Decryption Module to provide the security. This model greatly maintains the key size up to 8bits.

A. Block diagram of SAV model

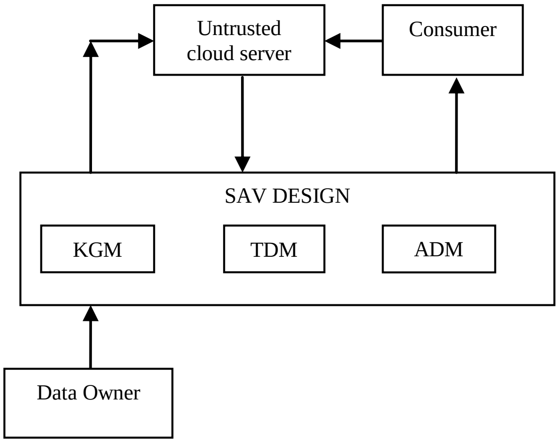

Fig. 1. Block diagram of SAV

B. Key Generation Module (KGM)

The Key Generation Module is responsible for the generation of keys used for the encryption and decryption. In this module our system enhances the key management polices while managing the users and keys. In this the public key generated is used for encrypting the text data uploaded by the data owner. The secret key generated by the KGM module is used for decrypting the encrypted data at the client end by the data consumer.

In the KGM module the encryption key bits are selected from a defined list such as 2nd bit, 3rd bit. 5th bit and 7th bit level. A proper bit pattern is selected and based on this bit pattern, list of secret keys are generated between two ranges of values. All the secret keys generated will be used for decryption with respect to the corresponding key pattern selected for encryption. This module also allows the specific choice of selection of some secret keys for the distribution to the authorized data consumers. This KGM module relates with the operation of file uploading process. File uploading to the cloud is usually done by the data owners registered and authorized to access a specific domain within the cloud. They can also mange the cloud once get authorized.

C. Transitional Encapsulation Module (TEM)

The Transitional Encapsulation Module is a major security model implemented in this system. The proposed additive Homomorphic operation is done under this module to give security for intermediate data. When a file is uploaded to the cloud by the data owner the file is subjected to pre-encryption, such that as part of implementation of AHE the file is pre-encrypted into two parts and each file is stored with the encrypted content in the cloud storage. This will reduce the security threats even if the cloud storage belonging to a un-trusted cloud server, because of two different versions of encrypted file is stored in the cloud storage. When a data consumer request for a PHR file to the data owner, the system receives the request and the TEM module works on the two version of the file requested file stored in the un-trusted cloud storage and begins the process of transiting additive Homomorphic encryption with these two versions and generates a newly encrypted file and stores in the trusted cloud.

D. Alternate Decryption Module (ADM)

The Alternate Decryption Module elaborates the process of decryption at clients end. The ADM is linked with the data consumers request for a PHR file. When the data consumer request is received by the system our proposed model, by analyzing the request executes TDM and encrypts the files and stores in the cloud storage. Once the original encrypted file is stored in the cloud server, the data consumer can download the file after data owner process the request. During the ADM process the data consumer gives a secret key for downloading and decrypting the file.

The secret key is processed by ADM and if there is a match with the encryption bit level used for encryption then the secret key is checked with the public key to find pattern matching. If there is a match then the encrypted text will be decrypted and downloaded to the data consumers.

E. Additive Homomorphic Encryption (AHE)

In this algorithm, the data to be shared securely will be divided in to two based on their ASCII values and then encrypt it before outsourcing to the cloud servers. This will secure the data from the untrusted cloud server itself. When the data is requested by the user, the data are retrieved and addition operation is performed on the splitted data and then decrypted to get the original data based on the key used.

F. SAV Algorithm

- Data owner encrypts m1 and m2 messages using the ASCII value with the help of public key generated from key generation module (KGM) of SAV algorithm.
- When a request is generated from the data consumer the AHE is done on m1 and m2 with the help of transitional encapsulation module (TEM) of SAV and send to the data consumer
- When the message processed at cloud is received by the data consumer, his credentials are verified and then he uses the secret key issued by the data owner to decrypt the message. It asks for a OTP for decryption which is created by the alternate decryption module (ADM) of SAV at the time of success of credential verification for the data consumer.
- The data consumer uses this OTP further to make the decryption as success.

G. SAV Process

The key generation algorithm is more protective. Two pair of keys are generated, one is used as public key (PK) and the second is used as secret key (SK) for decryption. When a data owner uploads a message ‘A’, it is resolved to its ASCII value,

A
↓
65

Then this ASCII value is splitted into two adjacent values say 33 and 32. This 33 and 32 are encrypted to its binary equivalents like

32→m1→00100000
33→m2→00100001

m1 and m2 are again encrypted using reverse masking and the final cipher text stored in untrusted cloud storage will be

C1→m1'→11011111
C2→m2'→11011110

When a data consumer request for the message ‘A’, the request is forwarded to the cloud storage. Based on the request the encrypted messages such as C1 and C2 are retrieved from the untrusted / third party cloud servers. The fetched C1and C2 are subjected to AHE then the resultant C3⇒ C1ϴC2 will be stored in the trusted cloud storage, ie, do inverse mask on

C1→ invmask () →00100000
C2→ invmask () →00100001

Then AHE is performed on C1 and C2

00100000
$\underline{00100001}$
C1 ϴ C2⇒ 01000001 ⇒ C3

Then using the public key the final Encapsulated Encryption is applied on the C3 using TEM, if the encrypted key used is at 2nd bit level, the 2nd bit in the

C3 is masked, ie
C3 → 00000001

□ This is the final encrypt message stored in the cloud server (trusted). This shows higher security of our original data.

When the client or data consumer sends the download request along with the secret key, then the encrypted C3 message is retrieved from the cloud server, the ADM work as follows C3 is then decrypted with 2nd bit level based on the decryption key given for decryption

C3 → 01000001

Again this C3 is subjected to damasking, i.e. from the group of bit values, a particular pattern of 8 bits is retrieved and is converted to the decimal equivalent, ie that

C3 → 65

Then this 65 value is compared with the ASCII index and the original value ie ‘A’ is restored and sends to Data consumer.

IV. Experimental Evaluation

The system used to test the above application must have a processor equals are greater than Core 2 Duo 2GHz, RAM 1GB, Hard Disk 80GB, Display of 1024 x768 resolution, and PS/2 keyboard and Mouse. The development and implementation of the proposed system can done with JSP as programming language under J2EE and SQL server is used as the database engine. The proposed system is tested and implementation under windows 7 environment. Our proposed system evaluates to best with the comparison of existing algorithms in terms of computation time and security. With the least bit size in encryption key length and manageable encrypted text the system provides unbreakable security to the user data at the cloud in un-trusted environments.

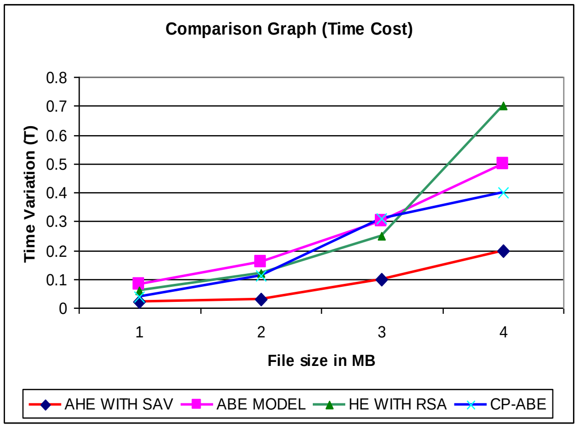

Fig. 2. Comparison graph of different AHE algorithm

V. Conclusion

To provide the effective sharing of personal health records in cloud, the system uses Secure Alternate Viable algorithm. It provides better key management and encryption technique based on additive Homomorphic Encryption technique. The proposed work is an innovative patient-centric framework and an alternate mechanism for data access control and is proposed for the PHR's stored in semi-trusted cloud servers.

To achieve fine-grained scalable data access control for PHRs, the proposed system leverages Additive Homomorphic Encryption (AHE) techniques to encrypt each patient's PHR file. Different from previous works in secure data outsourcing, this system focuses on an alternate secure model to confuse the un-trusted cloud servers or anonymous access controllers while accessing the records.

REFERENCES

[1] M. Li, S. Yu, K. Ren and W.Lou, "Scalable and secure sharing of Personal Health Records in Cloud Computing using Attribute based Encryption", IEEE Transaction on Parallel and Distributed Systems,vol. 24, no. 1., 2013.

[2] Pieter Van Gorp, Marco Comuzzi, "Lifelong Personal Health Data and Application Software via Virtual Machines in the Cloud", IEEE Journal of Biomedical and Health Informatics, vol. 0, no.0, 2012.

[3] Z. Erkin, Member, IEEE, T. Veugen, T. Toft, and Reginald L. Lagendijk, Fellow, IEEE, "Generating Private Recommendations EfficientlyUsing Homomorphic Encryption and Data Packing", IEEE Transactions on Information Forensics and Security, vol. 7, no.3., 2012.

[4] C. He, X. Fan, and Ye Li*, Member, IEEE, "Toward Ubiquitous Healthcare Services with a Novel Efficient Cloud Platform", IEEE Transactions on Biomedical Engineering, vol. 60, no.1. 2013.

[5] W. Wang, Y. Hu, L. Chen, X. Huang, B. Sunar Worcester Polytechnic Institute, "Exploring the Feasibility of Fully Homomorphic Encryption", IEEE Transactions on Computers, Digital Object Indentifier 10.1109/tc.2013.154., 2013.

[6] L. Guo, Student Member, IEEE, C.hang, Member, IEEE, J. Sun, Member, IEEE, Y. Fang, Fellow, IEEE, "A Privacy-Preserving Attribute-based Authentication System for Mobile Health Networks", IEEE Transactions on Mobile Computing, 2013.

[7] Bharath K. Samanthula, Y. Elmehdwi, G. Howser,S. Madria*,"A secure data sharing and query processing framework via federation of cloud computing", Information Systems, 2013.

[8] X.Yi, M.Golam Kaosar, R.Paulet, and E.Bertino, Fellow, IEEE, "Single-Database Private Information Retrieval from Fully Homomorphic Encryption", IEEE Transactions on Knowledge and Data Engineering, VOL. 25, NO. 5, 2013.

[9] Huang Lin, Jun Shao, Chi Zhang, and Yuguang Fang, Fellow, IEEE, "CAM: Cloud-Assisted Privacy Preserving Mobile Health Monitoring", IEEE Transactions on Information Forensics and Security, VOL. 8, NO.6, 2013.

[10] Z. Wang, G.Sun, D.Chen, "A new definition of homomorphic signature for identity management in mobile cloud computing", Journal of Computer and System Sciences, 2013.