Secure Data Sharing In Public Cloud Using Mediated Certificate less Encryption

Savitha N[1], Dr.S.V.M.G.Bavithiraja[2], M.Balachandar[3],
[1]*PG Scholar, Dept. of CSE, Sri Eshwar College of Engineering, Coimbatore, India*
[2]*Professor, Dept. of CSE, Sri Eshwar College of Engineering, Coimbatore, India*
[3]*CSM, Sistema Shyam,Teleservices*

***Abstract*- A mediated Certificateless encryption scheme without pairing operations is used for securely sharing sensitive information in public clouds. Mediated Certificateless Public Key Encryption (mCL-PKE) solves the key escrow problem in identity based encryption and certificate revocation problem in public key cryptography. mCL-PKE scheme applies to construct a solution to the problem of sharing sensitive information in public clouds. In this system, the data owner encrypts the data while uploading using the cloud generated users public keys based on its access control policies and uploads the encrypted data to the cloud. Cloud partially decrypts the encrypted data for the users after successful authorization. Users fully decrypt the partially decrypted data using their private keys generated while encryption. The confidentiality of the content and the keys is preserved with respect to the cloud, because the cloud cannot fully decrypt the information. Only data owner can encrypt and decrypt the data using keys. Inorder to improve the efficiency of encryption at the data owner, mCL-PKE scheme is used and the overall cloud based system evaluate its security and performance.**

***Index Terms*- Public cloud, mCL-PKE, Encryption, Decryption.**

I. INTRODUCTION

The cloud is employed as secure data storage and a key generation center. The data owner encrypts the sensitive data using the cloud generated users public keys based on its access control policies and uploads the encrypted data to the cloud. Data confidentiality is the main issue to solve for widespread adoption of cloud services. Shared sensitive data must be strongly secured from unauthorized accesses.

To assure the Confidentiality of sensitive data stored in public clouds, the data is encrypted before uploading it to the cloud. The cloud does not know the keys used to encrypt the data, the confidentiality of the data from the cloud is assured. Many organizations are required to implement fine grained access control to the data,

the encryption mechanism should be able to Support fine-grained encryption based access control. A normal approach used to support fine grained encryption based access control to encrypt different sets of data items to which the same access control policy.

The key derivation based approaches reduce the number of keys to be managed, symmetric key based mechanisms in general have the problem of high costs for key management. In order to reduce the overhead of key management, an alternative is to use a public key cryptosystem. A traditional public key cryptosystem requires a trusted Certificate Authority (CA) to issue digital certificates that bind users to their public keys.

Because the CA has to generate its own signature on each user's public key and manage each user's certificate, the overall certificate management is very expensive and complex.

II. METHODS

A. Attribute Based Encryption

Attribute Based Encryption (ABE) allows to encrypt each data item based on the access control policy applicable to the data. In addition to the key escrow problem, ABE has the revocation problem as the private keys given to existing users should be updated whenever a user is revoked. In order to address the key escrow problem Certificateless Public Key Cryptography (CL-PKC) is used. Only the data owner had the right modify the content, delete or made changes in the data. There is no need

to have a copy of the file. It can be downloaded whenever the user needs by using the private key. The user is able to decrypt using its own private key and an intermediate key issued by the data owner.

A typical approach in Fig 1 is used to support fine grained encryption based access control is to encrypt different sets of data items to which the same access control policy applies with different symmetric keys and give users either the relevant keys [4], or the ability to derive the keys. Even though the key derivation based approaches reduce the number of keys to be managed, symmetric key based mechanisms in general have the problem of high costs for key management. In order to reduce the overhead of key management, an alternative is to use a public key cryptosystem. Fig 1 shows that a traditional public key cryptosystem requires a trusted Certificate Authority (CA) to issue digital certificates that bind users to their public keys. Because the CA has to generate its own signature on each user's public key and manage each user's certificate, the overall certificate management is very expensive and complex.

To address such shortcoming, Identity-Based Public Key Cryptosystem (IB-PKC) was introduced, but it suffers from the key escrow problem as the key generation server learns the private keys of all users. Attribute Based Encryption (ABE) allows one to encrypt each data item based on the access control policy applicable to the data. In addition to the key escrow problem, ABE has the revocation problem as the private

keys given to existing users should be updated whenever a user is revoked.

Moreover, the scheme only achieves Chosen Plaintext Attack (CPA) security. As pointed out in [3], CPA security is often not sufficient to guarantee security in general protocol settings.

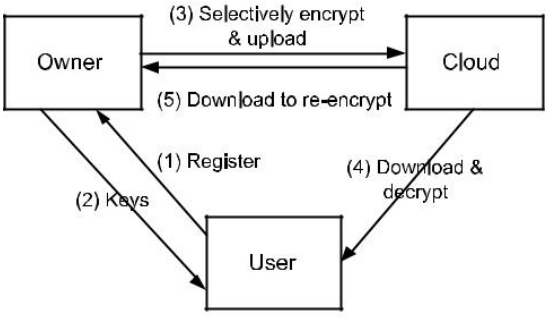

Fig.1. Symmetric key based fine-grained encryption.

CPA is not sufficient for many applications such as encrypted email forwarding and secure data sharing that require security against Chosen Ciphertext Attack (CCA).

B. Mediated Certificateless Public Key 0Encryption

Mediated Certificateless Public Key Encryption (mCL-PKE) scheme used for encrypt and decrypt the data uploaded to the cloud without pairing operations. Bilinear pairings is most widely used in all CL-PKC and also they are computationally expensive. It reduces the computational overhead by using a pairing free approach. Semi trusted security mediator is used to reduce the computation costs for decryption at the users and it partially decrypts the encrypted data before the users decrypt. The security mediator acts as a policy enforcement point as well and supports instantaneous revocation of compromised or malicious users.

Compared to symmetric key based mechanisms, our approach can efficiently manage keys and user revocations. In symmetric key systems, users are required to manage a number of keys equal to at least the logarithm of the number of users, whereas in our approach, each user only needs to maintain its public/private key pair. Revocation of users in a typical symmetric key system requires updating the private keys given to all the users in the group. Whereas private keys of the users are no need to be changed.

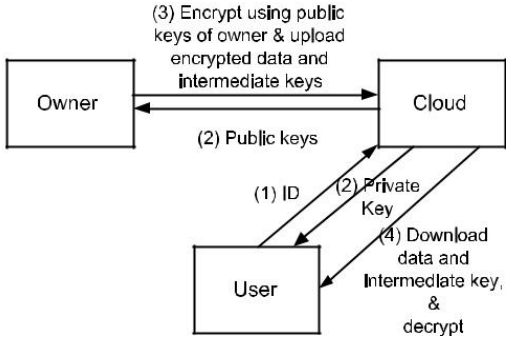

Fig. 2. CL-PKE with intermediate keys based fine-grained encryption.

Based on mCL-PKE, the confidentiality of data stored in public clouds is assured while enforcing access control requirements. The five entities used are 1)data owner, 2) users, 3)Security Mediator (SEM), 4)Key Generation Center (KGC), and 5)Storage service (Fig. 2).

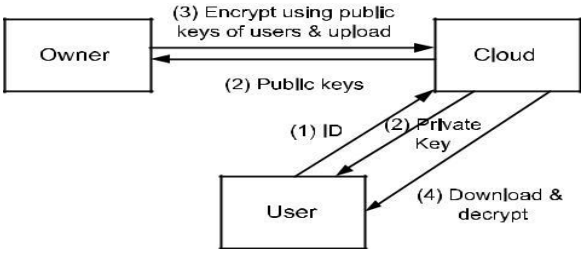

Fig. 3. CL-PKE based fine-grained encryption.

The SEM, KGC and the storage service are semi-trusted and reside in a public cloud. Although they are not trusted for the confidentiality of the data and the keys, they are trusted for executing the protocols correctly. The data owner encrypts a symmetric data using encryption key and encrypts the data items using symmetric encryption algorithm based on access control policy. Then, data owner uploads encrypted data items and the encrypted data encryption key to the cloud. Notice that a major advantage of our approach compared to conventional approaches is that the KGC, which is the entity in charge of generating the keys, resides in a public cloud. Thus, it simplifies a task of key management for organizations.

Users private key consists of a secret value chosen by the user and a partial private key generated by the KGC. From fig 2 unlike the CL-PKE scheme, the partial private key is securely given to the SEM, and the user keeps only the secret value as its own private key in the mCL-PKE scheme. So, each user's access request goes through the SEM which checks whether the user is revoked before it partially decrypts the encrypted data using the partial private key. It does not suffer from the key escrow problem, because the user's own private key is not revealed to any party. It should be noted that neither the KGC nor the SEM can decrypt the encrypted data for specific users. Moreover, since each access request is mediated through the SEM, our approach supports immediate revocation of compromised users.

The data owner has to encrypt the same data encryption key multiple times, once for each user, using the users' public keys. To address this shortcoming, we introduce an extension of the basic mCL-PKE scheme. Our extended mCL-PKE scheme requires the data owner to encrypt the data encryption key only once and to provide some additional information to the cloud so that authorized users can decrypt the content using their private keys. Fig. 3 gives a high level view of the extension. Proxy Re-Encryption (PRE) is by which the data encryption key is encrypted using the data owners public key. Later can be decrypted by different private keys after some transformation by the cloud which acts as the proxy. However, in this extension, the cloud simply acts as storage and does not perform any transformation. Instead, the user is able to decrypt using its own private key and an intermediate key issued by the data owner.

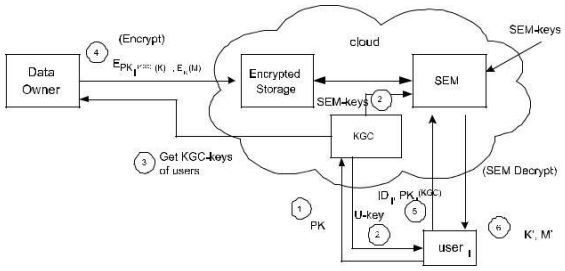

Fig. 4. The overall system.

Fig 4 represents the overall system of encryption and decryption.

III.SECURE CLOUD STORAGE

mCL-PKE scheme consists of three entities: data owner, cloud, and users. The data owner possesses sensitive content that it wants to share with authorized users by storing it in the public cloud and requesting the cloud to partially decrypt the encrypted content when users request the data. The cloud consists of three main services: an encrypted content storage; a key generation center (KGC), which generates public/private key pairs for each user and a security mediation server (SEM), which acts as a security mediator for each data request and partially decrypts encrypted data for authorized users. The cloud is trusted to perform the security mediation service and key generation correctly, but it is not trusted for the confidentiality of the content and key escrowing. It allows one to have most of the key generation and management functionality deployed in the untrusted cloud as our mCL-PKE scheme does not have the problem of key escrowing and thus the KGC is unable to learn the full private keys of users.

Our scheme consists of five phases: (1) Cloud set up; (2) Identity token issuance; (3) Identity token registration; (4) Data encryption and uploading; (5) Data view and Decryption; (6) Encryption evolution management.

2.1Cloud Set Up

The KGC in the cloud runs the SetUp operation of the mCL-PKE scheme and generates the master key MK and the system parameters params. It should be noted that this setup operation is a one-time task.

2.2 Identity Token Issuance

IdPs are trusted third parties that issue identity tokens to Users based on their identity attributes. It should be noted that IdPs need not be online after they issue identity tokens.

2.3 Identity Token Registration

Users register their token to obtain secrets in order to decrypt the data when it allowed to access. Users register their tokens related to the attribute conditions in ACC with the Owner, and the rest of the identity tokens related to the attribute conditions in ACB/ACC with the Cloud.

When Users register with the Owner, the Owner issues them into two sets of secrets for the attribute conditions in ACC that are also present in the sub ACPs in ACPB Cloud. The Owner keeps one set and gives the other set to the Cloud. Two different sets are used in order to prevent the Cloud from decrypting the Owner encrypted data.

2.4 Data Encryption and Uploading

The Owner first encrypts the data based on the Owners sub ACPs in order to hide the content from the Cloud and then uploads them along with the public information generated by the AB-GKM KeyGen algorithm and the remaining sub ACPs to the Cloud.

The Cloud in turn encrypts the data based on the keys generated using its own AB-GKM KeyGen algorithm. The AB-GKM KeyGen at the Cloud takes the secrets issued to Users and the sub ACPs given by the Owner into consideration to generate keys.

2.5 Data View and Decryption

Users download encrypted data from the Cloud and decrypt twice to access the data. First, the Cloud generated public information tuple is used to derive the OLE key and then the Owner generated public information tuple is used to derive the ILE key using the AB-GKM Key Gen algorithm. These two keys allow a User to decrypt a data item only if the User satisfies the original ACP applied to the data item.

2.6 Encryption Evolution Management

Over time, either ACPs or user credentials may change. Further, already encrypted data may go through frequent updates. In such situations, data already encrypted must be re-encrypted with a new key. As the Cloud performs the access control enforcing encryption, it simply re-encrypts the affected data without the intervention of the Owner.

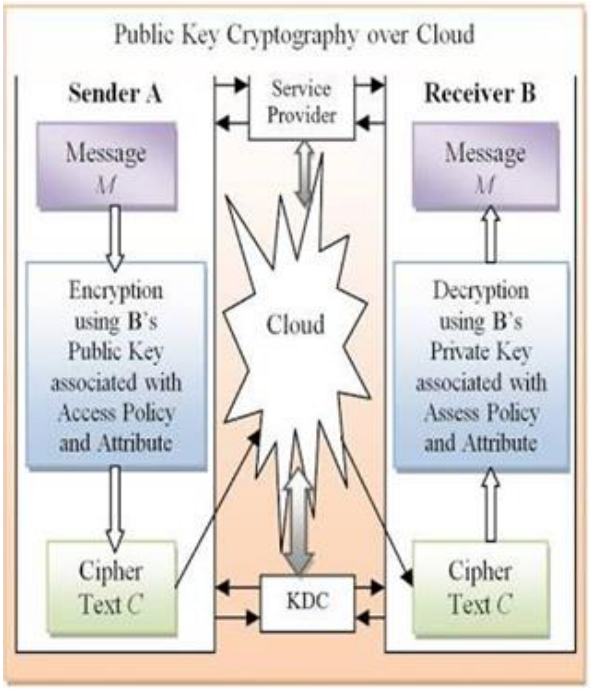

Fig. 5. Public Key Cryptography over Cloud

IV.IMPROVED SECURE CLOUD STORAGE

The data owner has to encrypt the same data encryption key multiple times for each authorized user. It can be a huge bottleneck at the data owner if many users are authorized to access the same data as the number of mCL-PKE encryptions is proportional to the number of authorized users. It provides an extension to our basic mCL-PKE scheme so that the data owner encrypts the data encryption key once for a data item and provides some additional information to the cloud so that authorized users can decrypt the content using their private keys. The idea is similar to Proxy Re-Encryption (PRE) where the content encrypted using the data owner's public key is allowed to be decrypted by different private keys after some transformation by the cloud which acts as the proxy. However, in our improved scheme, the cloud simply acts as storage for the proxy keys,

referred to as intermediate keys, and gives these keys to users at the time of data requests.

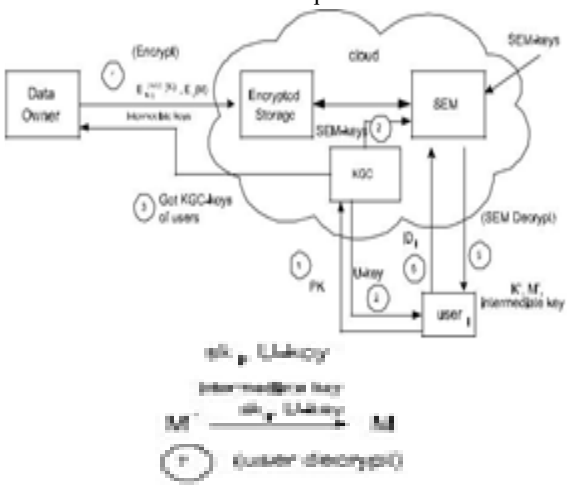

Fig. 6. The overall system with immediate keys.

Fig. 6 represents the overall system with the utilization of intermediate keys. The phases in this approach are very similar to those of the basic approach except for the following differences.

•During the data encryption and download phases, the data owner downloads the public keys of users to generate the intermediate keys as shown above. Unlike the basic approach, the data owner encrypts each data item only once using a random symmetric key K and then mCL-PKE encrypts K using its public key. The data owner uploads the encrypted data along with the intermediate keys to the cloud. The encrypted data is stored in the storage service in the cloud and the intermediate keys are stored at the SEM in the cloud.

•During the data retrieval and decryption phases, upon successful authorization, the SEM partially decrypts the data encrypted using the data owner's public key as input to the SEM-decryption operation of the basic mCL-PKE scheme, and gives the partially decrypted data along with the intermediate keys. The intermediate keys along with private keys allow users to fully decrypt the partially decrypted data using User-Decrypt operation of the basic mCL-PKE scheme.

V.EXPERIMMENTAL RESULTS

A solution for secure data sharing in public cloud is achieved by using mCL-PKE. An mCL-PKE algorithm was formulated, along with a multiple users in public cloud and corresponding policy evaluation mechanism. In addition, an approach for representing and reasoning about our proposed model. Encryption and decryption of a system that helps users automates the privacy policy settings for their uploaded files. The private and public key system provides a comprehensive framework to infer privacy preferences based on the information available for a given user. It proves that a practical tool that offers significant improvements over current approaches to privacy.

S.No	Efficient Data sharing Algorithm	Secure data sharing in public cloud
1	Expensive pairing operations.	No pairing operations.
2	Data owner keeps a copy of the data.	No need to maintain a copy of data. It can be easily downloaded whenever they need by using their private keys.
3	Need to establish private communication channels with the users to share the data.	No need of private communication channels. Data may shared based on users request by using one time password.
4	Any user can view and access those data.	Only approved user can download and access.

Tab.1. Comparison of Efficient data sharing and Secure data sharing in public cloud using mediated Certificateless encryption.

Tab 1 explains about Comparison of Efficient data sharing and Secure data sharing in public cloud using mediated Certificateless encryption.

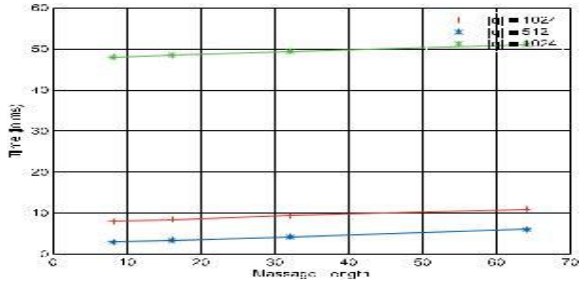

Fig. 7. Basic encryption

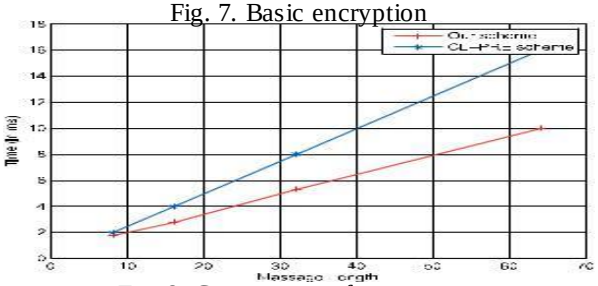

Fig. 8. Comparison of encryption.

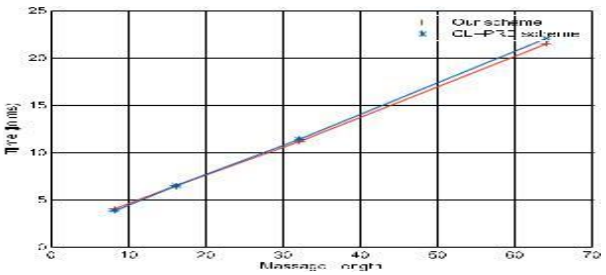

Fig.9. Improved Scheme.

Fig. 7 shows the time required to perform the encryption operation in the mCL-PKE scheme for different message sizes. Since our scheme does not use pairing operations, it performs encryption efficiently. As can be seen from the graph Fig.8 , the encryption time increases linearly as the message size increases. As the bit length increases, the cost increases non-linearly since the encryption algorithm performs exponentiation operations.

In the improved scheme the data owner performs only one encryption per data item and creates a set of intermediate keys that allows authorized users to decrypt the data. In Fig. 9, we compare the time to perform encryption and decryption in the basic scheme and the improved scheme as the number of users who can access the same data increases from 10 to 50. It is evident from the graph that as more users are allowed to access the same data item, the better the improved scheme performs compared to the basic scheme. The cost of the basic scheme is high since the encryption algorithm is executed for each user.

VI.CONCLUSION

The first mCL-PKE scheme without pairing operations provided its formal security. mCL-PKE solves the key escrow problem and revocation problem. Using the mCL-PKE scheme as a key building block, it provides an improved approach to securely share sensitive data in public clouds. This approach supports immediate revocation and assures the confide entiality of the data stored in an untrusted public cloud while enforcing the Access Control Policies of the data owner. Proposed System shows that the efficiency of basic mCL-PKE scheme and improved approach for the public cloud. Further, for multiple users who satisfies the same access control policies, this improved approach performs only a single encryption of each data item and reduces the overall overhead at the data owner.

REFERENCES

[1] Su Peng, Fucai Zhou, Qiang Wang, Zifeng Zu, and Jian Zu, “Identity Based Public Multi-Replica Provable Data Possession,” in Proc. Of IEEE Access, November 22,2017.

[2] You Zhou and Liang Min Wang, “SDS2: Secure Data-Sharing Scheme for Crowd Owners in Public Cloud Service” in Proc. Of IEEE Second International Conference on Data Science in Cyberspace, 2017.

[3] Chetan Gudisagar, Bibhu Ranjan Sahoo, Sushma M, Jaidhar C D, “Secure Data Migration between Cloud Storage Systems,” in Proc. Of ICACCI IEEE,2017.

[4] Xiaojie Niu, “Fine-grained Access Control Scheme Based on Cloud Storage” in Proc. Of International Conference on Computer Network, Electronic and Automation, 2017.

[5] Samundiswary.S, Nilima M Dongre, “Public Auditing for shared data in cloud with safe user revocation” in Proc. Of International Conference on Electronics, Communication and Aerospace Technology ,2017.

[6] A.Praveena, Dr. S.Smys, “Ensuring Data Security in Cloud Based Social Networks” in Proc. Of International Conference on Electronics, Communication and Aerospace Technology,2017.

[7] Savitha N, Dr. S.V.M.G. Bavithiraja, Shanthi T, “A Survey on Secure Data Sharing in Public Cloud Using Mediated Certificateless Encryption” in Proc. Of International Journal of Computer Science Engineering and Technology, Volume 8, Feb 2018.

[8] Hisham Abdalla,Xiong Hu, Abubaker Wahaballa, Nabeil Eltayieb,Mohammed Ramadan1, Qin Zhiguang, “Efficient Functional Encryption and Proxy Re-cryptography for Secure Public Cloud Data Sharing” in Proc. Of IEEE ICOACS,2016.

[9] Mohamed Yoosuf and Mohamed Nabeel, “An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds,” in Proc. Of IEEE Transactions on Knowledge and Data Engineering,Volume 26,2014.

[10] JueeliDangur, S.M.Jaybhaye,“Framework for Secure Data Sharing In Dynamic Group Using Public Cloud”, in Proc. Of “International

Conference on Computing, Analytics and Security Trends,2016.

[11] Mrs.Priyadharsini.V, Dr.S.V.M.G. Bavithiraja, M.Mohanapriya and Mr.M.Balachandar, "Performance Evaluation of 4G Uplink MAC Scheduling Algorithms – A Review", in Proc. Of International Journal of Advanced Engineering Recent Technology, Volume 17, Issue 1, March 2017.

[12] K.Selvakumar, Dr.S.V.M.G.Bavithiraja, Dr. C.Gunavathi and Mr.M.Balachandar, "Application of Cloud Computing in Aerospace and Defense", in Proc. of International Journal of Advanced Engineering Recent Technology, Volume 17, Issue 1, March 2017.

[13] Arulkumar, C. V., and P. Vivekanandan. " An intelligent technique for uniquely recognising face and finger image using learning vector quantisation (LVQ)-based template key generation." in Proc. Of International Journal of Biomedical Engineering and Technology, 3/4, 237 – 249, 2018.

[14] C.V.Arulkumar, K. Jeyakumar, M. Malarmathi, T. Shanmugapriya," Secure Communication in Unstructured P2P Networks based on Reputation Management and Self Certification", in Proc. Of International Journal of Computer Applications (0975 – 8887) Volume 44– No15, April 2012.

[15] Kaitai Liang, Man Ho Au, Joseph K. Liu, Willy Susilo, Duncan S. Wong, Guomin Yang, Tran Viet Xuan Phuong, and Qi Xie "A DFA-Based Functional Proxy Re-Encryption Scheme for Secure Public Cloud Data Sharing K" in Proc.Of IEEE Transactions On Information Forensics And Security, Vol. 9,October 2014.

[16] Kaiping Xue and Peilin Hong, "A Dynamic Secure Group Sharing Framework in Public Cloud Computing" in Proc. Of IEEE Transactions on Cloud Computing, 2014.

[17] Shahina K M, Deepak Lal , "A Survey on Secure Data Sharing Methods in Public Cloud Computing" in Proc. Of International Journal of Science, Engineering and Technology Research, Volume 5, Issue 01, January 2016.

[18] Ramyasree Nandagiri and Dinesh Chandrasehkaran, "Secure Data Sharing Using Certificate less Encryption for Providing Efficiency in Public Clouds" in Proc. Of International Research Journal of Engineering and Technology, Volume 02,Issue 09, December 2015.