Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.017

IJCSMC, Vol. 5, Issue. 11, November 2016, pg.44 – 51

RE-OTP KEY MANAGEMENT MECHANISM FOR THE CLOUD DATA SECURITY

Rishi Kumar Sharma[1], Dr. R.K.Kapoor[2], Pavan Kumar Sharma[3]
Research Scholar, Computer Science, AISECT University, Bhopal, India[1]
Associate Professor, Computer Science, NITTTR, Bhopal, India[2]
Assistant Professor, Computer Science, JLN College Bhopal, India[3]
[1] rishi.rishi1526@gmail.com; [2] rkkapoor@nitttrbpl.ac.in; [3] pa_ka_sh@yahoo.com

Abstract— Cloud storage is one of the most popular services techniques in cloud computing environment. A crucial problem to be addressed in the cloud storage system concerns the security mechanism of protecting the private data and sharing data with other users. Cloud storage services are inherently insecure as the management technique of the data in the cloud storage is controlled by third parties beyond the reach of the data owner. The secrets keys used for obfuscation are stored in a secure location (cloud location) while obfuscated data is stored in the cloud. In this approaches, the data is as secure as its corresponding keys. However, this still brings a challenging issue where a user needs to manage a large number of (secret) keys in such a way that they are guarded against all types of adversaries, and should be as highly available as cloud storage services. In this paper, we present a Re-key management technique to protect the security of user data and enforce the data access control. In our scheme is suitable for personal cloud storage and especially available to enterprise users to management their key of different kinds of resources. We describe the heritage system model, the details of the key management service and a prototype implementation.

Keywords— Re-rekeys management; Re key model; Over-Encryption in the cloud

I. INTRODUCTION

In recent years, with the growing popularity of cloud services provided by enterprises, the emergence of a variety of Security incidents have caused people's doubts and concerns of cloud service security.

As an emerging computing technique and model, cloud computing gets the attention of domestic and foreign researchers. Cloud computing puts a large amount of computing resources, storage resources, software resources together, formed the scale huge Shared virtual resource pool for remote users with various services. However, the development of cloud computing is faced with many problems, and the most important the data safety problem.

In the traditional network, the security of the computer and network is ensured by the password technology such as identity authentication, data encryption and so on. Cloud computing is an evolutionary new cloud Re-key model for distributed computing consisting of centralized data and decentralized datacenters that provide resources for massively scalable units of computing.

In cloud computing environment, the key security is also the key to the whole cloud system, the data encryption, virtual machine security isolation; identity authentication and access control technology all need the technical support of key management system. Therefore it is necessary to design are liable rekey management scheme (Re-key model) for cloud environment.

In cloud computing environment, users can store their data on Cloud which can save their personal storage resources greatly and data owners can achieve their data with a cloud client anytime, anywhere. However, Cloud storage is faced with a series of safety problems.

II. RELATED WORK

[1] Now researchers' study of cloud computing security is still in preliminary stage, the industry has not yet formed the corresponding standards, and the study of cloud environment key management also remains in the theoretical stage. In [1]the authors introduce a PKI certificate scheme to secure cloud environment application program of encryption key, symmetric keys are stored by the public key of the certificate. Reference [2] presents a scheme that binds the key with files for personal storage. The scheme is easy to manage and maintain key update, however, the program requires the user's login password regularly updated, or if the password leaks, file security cannot be guaranteed. Kumar encrypts the data stored in the cloud with elliptic curve encryption [3]. Nepal [4]puts forward the scheme that divides the data into several sections, then randomly generates different keys for each piece of data and uses these keys to encrypt data.

In [5] the authors present the over-encryption management to protect outsourced resources. The proposed scheme uses two layer model. Resources owner encrypts the resources first before uploading the data to the server. And the server will encrypt the data the second time before users viewing the resources. In this paper, we apply the over-encryption scheme into the cloud computing environment. Data owner needs to encrypt his data before uploading the data to the cloud. The cloud server should encrypt the data again before other users downloading the data. Reference [6] has proposed the group key management based on key hyper graphs which is scalable to large groups. The key hyper graph is used in our scheme to manage the keys of the

system. The key management based on key hyper graph is a group key management which is available for large-scale application environments.

There is a clear need for a scalable and efficient key management solution for cloud computing systems, but so far it has not been fully addressed in commercial cloud systems. The Key Management Interoperability Protocol (KMIP) addresses the issue of interoperability of key management services, but fails to account for the unique scalability potential in cloud systems and the performance problems that can result. Open ID is an open-source Single Sign-On (SSO) solution that permits a single login to access different sites and resources, but effectively the same password is used to access multiple sites, with no fine-grained access control.

A traditional approach to communication security has be encentralized key management, which requires public-key certificates to be generated by the authority and deployed to all users before communication can occur. In a highly scalable system, the authorization server may become overloaded as a result of this responsibility. Security enforcement based on monitoring of user behavior can mitigate these performance concerns, such as in Trust Cube [7], but the cloud provider must been trusted with aggregated data on user contexts and activities, thus relaxing the trust model. In Certificate less Public Key Cryptography (CLPKC) [8], the Key Generation Centre (KGC)residing in the cloud does not have access to users' private keys, but the KGC would need to ensure that partial private keys would be delivered securely to the right users using some secure, or out-of-band, transport. Broadcast encryption maybe employed, in which the key manager generates symmetric keys for multiple users, but whenever the membership changes, then new keys must be rebroadcast to all users, which is an unrealistic proposition in a highly scalable system.

The high turnover of cloud user membership poses a great challenge; expensive **re-keying operations** are normally required whenever group membership changes. In the Logical Key Hierarchy [9] scheme, the processing time per request scales linearly with the logarithm of group size, and the signing of rekey messages increases server processing time by an order of magnitude. Another approach is distributed key management, where multiple distributed public key generators (PKGs) hold shares of a master key using the concept of threshold decryption [10], or portions of a private key are distributed among users [11]. The problem with these approaches is that a user must assemble a key from multiple sources, resulting in expensive communication sessions.

III. KEY GENERATION SYSTEM

In a cryptographic system, the key is generated in two ways. One is that the key is generated with the use of cryptographic algorithms. The other is that the key is generated with the use of auto random number generation. Once the key is generated, it is sent to the user through the provider then user send key then once again the key is generated, it is sent to the user through the provider. The user encrypts the message with the use of key and sends it to the provider then . The provider stores the encrypted data in the cloud. The Key Management Mechanism maintains all the information as shown in Table 1.

Table 1 maintains the information of cloud user, cloud service provider (CSP), key and the time utilized for the generation of key. Every time the user requests the provider for the key and Re-

key, the provider sends the request to the key generation system. Once the key is generated, it is sent to the user through service provider. At the same time the Re-key management mechanism maintains the information about the cloud user, service provider, the key and the time taken for generation.

TABLE I. GENERATED KEYS WITH CORRESPONDING PROVIDER AND USER

S_No	Cloud_User	CSP	Key	Key	Time (milliseconds)	Time (milliseconds)
1	Clouduser1	CSP1	K1	R-K1	T1	R-T1
2	Clouduser2	CSP2	K2	R-K2	T2	R-T2
3	Clouduser3	CSP3	K3	R-K3	T3	R-T3
4	Clouduser4	CSP4	K4	R-K4	T4	R-T4

IV. COMPONENTS IN KEY MANAGEMENT

As discussed earlier, the key management mechanism is more important to keep the data more confident and secure. There are few necessary components available in key management mechanism that is to be considered very necessary. Secure Key management is the management of cryptographic keys in a cryptosystem. Secure Key management deals with the generation, replacement, storage, use, and exchange of keys.

Secure Key management includes cryptographic protocol design, user procedures, key servers, and other relevant protocols. Secure Key management concerns keys at the cloud user level, either between cloud users or systems [7].

A. Key Generation

The cryptographic systems in modern technology include symmetric-key algorithms (such as DES and AES) and public key algorithms (such as RSA). The user encrypts the data with the public key. Only the provider of the private key decrypts this data. The simplest method to read encrypted data is a force fully attacks, meaning just attempting every number, up to the length of the key. Therefore, it is important to use sufficiently longer key length since longer keys take longer time to attack, resulting brute force attack almost impractical.

B. Key Destructions

The key destruction explains the deletion of the keys from the table. Keys, when no longer needed, must be destroyed in a secured manner. And also when the key is once used for the encryption, it must not be used by another user. Keys when not used within the stipulated time frame then it makes no relevance and so the key must be removed from the table.

C. Key Establishment

The Key establishment is done in three ways namely, key pre-distribution, key distribution, and key agreement. In this stage, the pre-distribution gives the concept of distributing keys to the user in reserve. The key distribution is of as and when the demands arise the key is distributed by the management. Key agreement is on the basis of cost factor between the user and the provider.

D. Key Storage

The keys must be stored securely to maintain the security in the process of communications. There are various techniques in use for this purpose. The most common technique is that an encryption application manages to keep the keys for the user and it depends on an access password to control the use of the key.

E. Key Change

The facility to change keys in all cryptographic systems is a must. For an example, the regular updates (planned) are done periodically. The key compromise, which is the unplanned one, will cause loss of data. Many systems are designed in such away that it is extremely difficult and expensive to change certain keys.

F. Key Usage

The length of the key use is the major issue. The keys must be frequently changed as the required efforts of the attackers are on the increase. The frequent change also limits the loss of information. The frequency of usage decreases as the frequency of key change increases. This happens especially when the attacker tries to trace the keys. Symmetric keys have been used for longer times since key exchange has become a difficult process. The symmetric keys must change with every data or interaction, so that only the intended data will become accessible even if the key is stolen, crypt analyzed, or socially engineered.

V. THE PROPOSE SCHEME

A framework is proposed to explain the Re-Key Management Mechanisms (Re-KMM), in this paper. Figure 1 gives the proposed framework. The user sends a request to the server asking for the key. The server asks for the key from the key management mechanism. The key management mechanism in turn asks the generator to generate the random key. The key is sent to the server and the details about the server and the user is stored in a table maintained by Re-Key management mechanism then again process recall and find valid user.

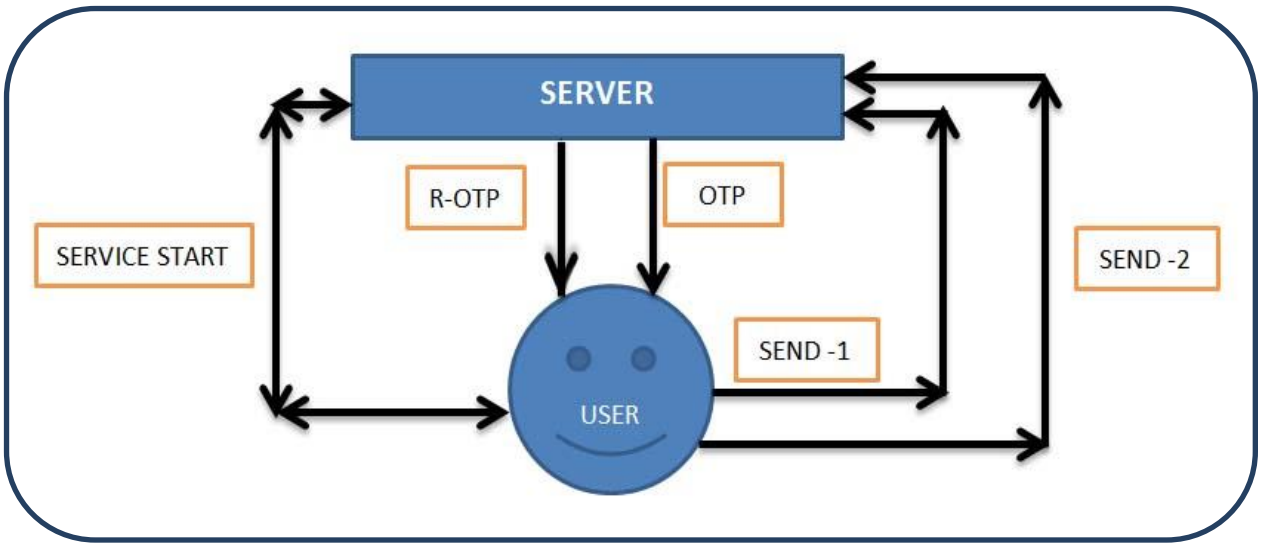

Fig. 1.Re-OTP SERVICE MODE

With the above-mentioned stages in the Re- key management mechanism, the Metadata table is maintained to protect the confidentiality of the data. The structure of the metadata table is managed by the Re-Key management mechanism. This structure maintains the table of logs information. Table shows sample representation of the metadata content maintained by the Re-key management mechanism. After a key is generated, the generated key and details are forwarded to Re-key management mechanism. Re-Key management mechanism stores the key with relevant information in the metadata.

TABLE II. STRUCTURE OF Re-KEY MANAGEMENT MECHANISM

S. No.	C1_USER1	C1_Service	key	Key_ Type	Key status	Re-key	Re-key_type	Re_Key status	Session
1	Cluser1	SR1	K1	Permanent/ Temporary	Alive/ Dead/ Withdrawn	R_K1	Permanent/ Temporary	Alive/ Dead/ Withdrawn	Sess1
2	Cluser2	SR2	K2	Permanent/ Temporary	Alive/ Dead/ Withdrawn	R_K1	Permanent/ Temporary	Alive/ Dead/ Withdrawn	Sess1
3	Cluser3	SR3	K3	Permanent/ Temporary	Alive/ Dead/ Withdrawn	R_K1	Permanent/ Temporary	Alive/ Dead/ Withdrawn	Sess1
4	Cluser4	SR4	K4	Permanent/ Temporary	Alive/ Dead/ Withdrawn	R_K1	Permanent/ Temporary	Alive/ Dead/ Withdrawn	Sess1

VI. CHALLENGES IN KEY MANAGEMENT

Though the key management mechanism is a safe guard technique to protect the confidentiality of the data, it has got its own challenges. The cloud users face several challenges while trying to control and manage the data encryption. They challenges are [12]:

Complex Management: As the numbers of users are on the increase in the cloud environment, the need for keys is also on higher demand. As the keys are large in quantity, managing the encryption of keys are in vast number. And so the complex scenario is visible.

Security Issues: The vulnerability of keys from outside hackers, and presence of malicious insiders and unauthorized users are in line to brute face attack. So the security issues are to be taken into consideration.

Data Availability: The ensuring data accessibility for authorized users is a big challenge.

Scalability: Supporting multiple databases, applications and its uses are big concern.

Governance: The defining of policies, access control and protection for data are to be carefully governed as the keys are in multiple accesses.

VII. PERFORMANCE EVALUATIONS

(1) **Data Confidentiality:** The over-encryption mechanism can ensure data confidentiality to protect data owner's privacy from being invading by the cloud server, since the cloud server is not always so credible. The second encryption by the cloud can protect the data from being attacked by someone who doesn't have the access to data through the open channel.

(2) **Access Control:** If the user who is not certified by the data owner sends the request to get the data, the cloud server will refuse his request for his ID is not in the table of the data. The rekeying strategies of the system can effectively prevent the data from the attacks of new users and the old users.

(3) The Re-key management based on hyper graph is particularly suitable for large enterprise users to store data shared with internal employees and customers. We use the multicast strategy in the process of file distribution significantly reduce the complexity of file encryption and distribution and improve the efficiency of the system.

VIII. CONCLUSION

In this paper, we discussed the key management mechanisms in order to protect the confidentiality of the data. We realize that the key management mechanism is more important than the key generation. The table is maintained to keep the track of the keys, the users and the providers. Various stages, many challenges, few techniques are discussed in this paper to understand the Re-key management mechanisms in a better way. The scheme is suitable for personal cloud storage and especially available to enterprise users to management their key of different kinds of resources. The Re- key management mechanism along with its components discussed here solves better solution and gives a new approach for the confidentiality of the data.

ACKNOWLEDGEMENT

This research work is partly supported by the AISECT University and Scope College of engineering.

REFERENCES

[1] Jindi G, Zishan D, Lei S. Research on Key Management Infrastructure in Cloud Computing Environment.[J]. International Conference on Grid & Cooperative Computing, 2010:404 - 407.

[2] Kumar A, Lee B GLee H J,et al, Secure storage and access of data in cloud computing[C].Proceedings of 2012International Conference on ICT Convergence (IC-TC) .NewYork: IEEE,2012:336 –339.

[3] Nepal S, FriedrichC , Henry L,et al,A secure storage service in the hybrid cloud[C].Proceedings of 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC) .NewYork: IEEE,2011:334 -335.

[4] Chen F H, Liu Y, Qi Y U. Key Management Scheme of Personal Cloud Computing [J]. Modern Computer, 2011.Personal Cloud Computing [J]. Modern Computer, 2011.

[5] Vimercati S D C D, Foresti S, Jajodia S, et al. Over encryption :management of access control evolution on out sourced data[C]// Very Large Data Bases2007:123-134.

[6] Yan DING, XianweiZHOU. Secure Group Communications Using Key Hyper graphs[J]. Journal of Computational Information Systems 8: 12 (2012) 5035–5042.

[7] R. Chow, M. Jakobsson, Y. Niu, E. Shi, J. Molina, R. Masuoka, andZ. Song, "Authentication in the clouds: a framework and its application to mobile users," in ACM Cloud Computing Security Workshop (CCSW),October 8, 2010 2010.

[8] S. S. Al-Riyami and K. G. Paterson, "Certificate less public key cryptography," Cryptology ePrint Archive, Report 2003/126, 2003, http://eprint.iacr.org/.

[9] C. K. Wong, M. Gouda, and S. S. Lam, "Secure group communicationsusing key graphs," IEEE/ACM Trans. Netw., vol. 8, pp. 16–30, February2000.

[10] D. Boneh and M. Franklin, "Identity-based encryption from the weilpairing," in Advances in Cryptology — CRYPTO 2001, ser. Lecture Notesin Computer Science, J. Kilian, Ed. Springer Berlin / Heidelberg, 2001,vol. 2139, pp. 213–229.

[11] J. Baek and Y. Zheng, "Identity-Based Threshold Decryption," in PublicKey Cryptography – PKC 2004, ser. Lecture Notes in Computer Science,F. Bao, R. Deng, and J. Zhou, Eds. Springer Berlin / Heidelberg, 2004,vol. 2947, pp. 262–276.

[12] "Security Policy and Key Management: Centrally Manage EncryptionKey". Slideshare.net. 2012-08-13.Retrieved 2013-08-06.

[13] Julian Jang-Jaccard,"Portable key management service for Cloud storage",Oct 2012, International conference on collaborative computing

[14] Kuei-Yi Chou,"An Efficient and Secure Group Key Management Scheme Supporting Frequent Key Updates on Pay-TV Systems", 2012 IEEE11th International conference on Trust, Security and Privacy in Computing

[15] Lee Badger Tim Grance Robert Patt-Corner Jeff Voas, " Cloud Computing Synopsis and Recommendations" in NIST Special Publication 800-146 ,May 2011

[16] S. AnahitaMortazavi, AlirezaNemaney Pour, Toshihiko Kato,"AnEfficient Distributed Group Key Management using Hierarchical Approachwith Diffie-Hellman and Symmetric Algorithm: DHSA",CNDS Feb 2011

[17] ENISA,"Algorithms, Key Sizes and Parameters Report,2013",recommendations version 1.0 – October 2013

[18] Y. Fan, L. Xiao-ping, D. Qing-kuan and L. Yan-ming, "A Dynamic Layering Scheme of Multicast Key Management," IEEE 5th International Conference on Information Assurance and Security (IAS '09), Xian, China,pp. 269-272, Aug. 2009

[19] Rajesh Ingle, G. Sivakumar, "EGSI: TGKA based Security Architecture for Group Communication in Grid", 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, pp. 34-42, 17-20 May, 2010.

[20] NIST, "Cloud Computing Synopsis and Recommendations", Specialpublication 800-146, May 2012.

[21] Yung-Wei Kao, Kuan-Ying Huang, Hui-Zhen Gu, Shyan-MingYuan,"uCloud: a user-centric key management scheme for cloud data protection", IET Journal, 28 January, 2013.

[22] JV Aghav,CVDeshpande, AS Shetye, SS Ghuge,"Towards a WebJDK: Extending openJDK 7 for client file system access over cloud" in Information & Communication Technologies (ICT), 2013, 11 April 2013.

[23] Sunil Dorwani J Aghav, BhalchandraBhutkar," NoSQL: Features, Classification and Comparison", IEECS-2013, ISBN : 978-93-83060-02- 3,21 April 2013.

