Volume 7, Issue 5, May 2017 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Cryptographic Tree and Its Key Management for Securing Outsourced Data in the Cloud

[1]Vairaprakash Gurusamy, [2]S. Kannan, [3]T. Maria Mahajan
[1, 2] School of IT, Madurai Kamaraj University, Tamilnadu, India
[3] Vivekananda Arts and Science College for Women, Tamilnadu, India

Abstract: ***Cloud computing from the last few years has grown from a promising business concept to one of the fastest growing segments of IT Enterprise. This unique paradigm brings about many new security challenges. This work studies the problem of ensuring the storage security to the data outsourced in the cloud which is accessed and controlled by multiple legitimate parties. For this, we propose a very practical approach of providing privacy to the data outsourced in the cloud environment which makes use of both Cryptographic Tree and its key management to secure the data outsourced in the cloud. This scheme also allows multiple legitimate parties to outsource their data to the cloud without making a compromise on security and better performance than existing ones.***
Keywords

Key management, cryptree, securing outsourced data, cloud

I. INTRODUCTION

Cloud computing security is an evolving sub-domain of computer security and network security. It refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in third-party data centers. Organizations use the Cloud in a variety of different service models SaaS, PaaS, and IaaS and deployment models Private Public, Hybrid, and Community.

There is a number of security issues/concerns associated with cloud computing but these issues fall into two broad categories: security issues faced by cloud providers and security issues faced by their customers as companies or organizations who host applications or store data on the cloud. The responsibility goes both ways, however: the provider must ensure that their infrastructure is secure and that their clients' data and applications are protected while the user must take measures to fortify their application and use strong passwords and authentication measures.

II. RELATED WORKS

Various authors have proposed various kinds of key management schemes. Among these, many of the existing works are on group key management in wireless sensor networks and some works are on cloud storage systems.

(M. Zhou et. al 2012) has proposed a tree based cryptographic key management scheme for data storage in the cloud. This method is based on tree-based scheme, where there is a root node which holds the master key and the users who come below the root node in the tree hierarchy. Here, there is a cloud provider who acts as a root node and there are data owners, sub-tree data manager and a user who stores data in the cloud system. So, in this work, there are multiple users with multiple levels of access rights. Each of them one encryption and two decryption keys. This method is a secure and flexible tree-based key derivation hierarchy and allows only the outsourcing user to access the data block located at a specific node while other data are encrypted with child keys. This method is very effective for hierarchy based user access control and key management system.

(D. Grolimund et. al. 2006) has proposed a cryptree method for granting access to files and folders based on file system's folder hierarchy system. They have proposed two cryptree methods: one for read access and another for write access. And they have presented their experimental results and findings.

(J.J.Jaccard et. al. 2012) has proposed a portable key management service for maintaining a large number of keys that are stored in a designated storage area. They have implemented their key management system that is based on a portable USB device, which the users need to carry with them all the time. Access is provided to a particular user when they plug in their USB device containing information about their access rights. The user cannot access his data when he has lost his USB device. This could lead to loss of information and in the worst case, the device may fall into wrong hands, and thus allowing an unauthorized person to access the data.

(T.Chih Hsia et. al. 2012) have used Lagrange interpolation method for key generation and encrypt it with Elliptic Curve cryptography. They claim that due to the use of Lagrange interpolation polynomial, the key generated is random. Thus there is no relationship between the generated keys. Also, due to the use of ECC, the attacker has to undergo Elliptic curve discrete logarithm problem to crack the key. They have proved that their system works in medical records

storage scenario, where each doctor holds the higher level key and the nurses hold the lower level key. The documents are split into several parts depending on the required level of privacy and these are put into a relationship structure into a server host or databases from which they can be accessed. This method can be effective when used for cloud based access mechanism structure. But it is more prone to internal attacks such as in the case where there is an insider who wants to gain access to a higher level decrypt key and uses it to maliciously modify the user's data.

(Shikha Sharma, C. Rama Krishna 2015) have proposed a method for group key management on a group with a hierarchy. Their key management system is based on Elliptic Curve Cryptography. Diffie-Hellman and Symmetric Algorithm are used for managing keys. When a new member joins in the group structure, that new member requests access keys from another user in his hierarchy level. This method is effective in the case where group management is the required focus of interest. When it comes to cloud systems, the user cannot be added without the proper permission from the cloud service provider or the cloud manager when in the case of a private cloud.

(H.Chung Chen, Anita Christiana 2014) has proposed a key management method in a role-based access control structure. The approach takes advantage of combining RSA and Akl Scheme. Akl is used for key generation and RSA for encryption/decryption phases. It can be implemented very easily as the key generation and derivation methods are simple and easy to use. This method is prone to two types of attacks: outsider attack and lower privilege user attack. Outsiders can attack at any level of the hierarchy and try to access their key. Also, any lower privilege user can try to attack an upper privilege user's keys and make any changes in the data or the key itself. The security of this method relies on RSA algorithm.

(Roney Thomas, Salim A 2014) have proposed a decentralized key management scheme. This system is controlled by a trusted subgroup controllers and two key generators. They claim to reduce the number of keys that are stored by the subgroup controller. This scheme uses Ciphertext policy Attribute based Encryption for the secure group by sharing between the members of the group. They have compared their proposed scheme with Lolus, an existing decentralized group key management scheme that was proposed by Mohammad Suvo Mittra.

(Amar R. Buchade, Rajesh Ingle 2014), have analysed the various key management methods for the cloud. Specifically, they have identified, compared and tested different symmetric and asymmetric key management methods. They have also identified the limitations of various methods. An analysis of the key management schemes for various scenarios also has been provided. They have concluded that symmetric key algorithms work best on cloud environment without affecting the basic characteristics of the cloud.

Sheng Xiao et. al. (2014) have argued about the reliability factor for maintaining a key. They suggest the necessity of including reliability requirement in key management scheme design. They have proposed a set of methods that an effective key management system must satisfy.

III. KEY MANAGEMENT

Keys are the basic entities of a cryptographic process. Therefore the key is a very important aspect of security. Cryptographic keys are used to keep data confidential from other users. Key management techniques involve key generation, distribution, storage, revoking and verifying keys (Amar R. Buchade, Rajesh Ingle 2014).

Principles of Key Management

Managing keys requires three considerations: They are

1. Storing keys
2. Keys must be protected but available when needed
3. Adequacy of key strength for the data to remain protected

Key Storage

Many organizations store key files on the same system, and often the same drive, as the encrypted database or files. While this might seem like a good idea if your key is encrypted, it is bad security. When the system fails and the key is not recoverable, having usable backup helps, but backup restores do not always work as planned.

Regardless of where you keep your key, encrypt it. Of course, now you have to decide where to store the encryption key for the encrypted encryption key. None of this confusion is necessary if you store all keys in a secure, central location. Further, do not rely solely on backups. Consider storing keys in escrow, allowing access to a limited number of employees ("key escrow"). Escrow storage can be a safe deposit box, a trusted third party, etc. Under no circumstances allow any one employee to privately encrypt your keys.

Key Protection

Encrypted keys protecting encrypted production data cannot be locked away and only brought out by trusted employees as needed. Rather, keep the keys available but safe. Key access security is, at its most basic level, a function of the strength of your authentication methods. Regardless of how well protected your keys are when not used, authenticated users (including applications) must gain access. Ensure identity verification is strong and aggressively enforce separation of duties, least privilege, and need-to-know.

Key Strength

A key strength is a very important aspect of a key management system.

IV. PROPOSED WORK

In our proposed work, we have implemented a group management system with each user given specific access rights and this scenario is based on private cloud system

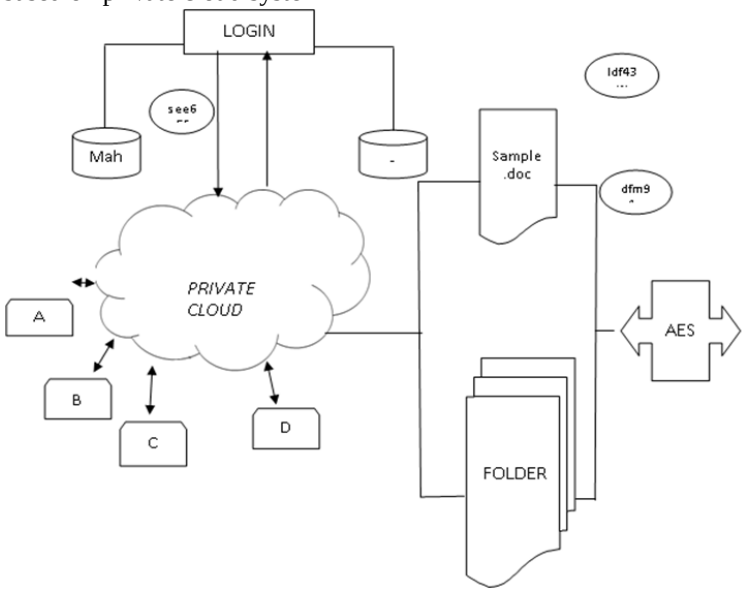

Figure 1. Flow diagram for Key Management

When a new user enters the cloud system, in this case, an organization, the user is given a random user ID at first. This ID is later updated by the cloud administrator based on the ID of the members present in the group. In the case of occurrence of more of two groups in the cloud, the administrator decides about which group the new user could be placed.

Access Rights

Three kinds of access rights are given to the user:

1. Private or read only access
2. Read and Write Access
3. Access for non- static or dynamic members of the cloud.

The private access data are mostly held by the head or higher level authorities of the organization. These data can be shared by them to the other users in case he needs them to access specific parts of the data, such as filling an online feedback or complaints form.

Read-write access is used by static members of the group. These kinds of users are allowed to access their data. They can also access the data that any higher official has granted access to them.

The third kind of access rights is for dynamic members of the group. This can be used for visiting members who need temporary access to the cloud for saving, viewing and modifying their data. They cannot access the data of other users of the group they exist or any other group under the particular cloud as a whole.

Figure 1 shows the overview of the proposed work. In this, when the user enters in the scheme, then the system generate the random number for every user. And it will generate the access key for every user who belongs to which group. After getting the access permission, the user can upload the files or folder with read or read/write permission. Suppose the file is read permission, the users can access the file with read permission. Otherwise, the user can access read / write permission. All the keys are generated, the keys are managed by cryptree concepts.

Key Generation

When a user uploads a file into the cloud, the key is generated for the file. Key generation is based on a MD5 algorithm. The key is hash value based on the file name and the timestamp.

The symmetric key algorithms work best on a cloud environment. (Amar R. Buchade, Rajesh Ingle (2014)), I have used AES encryption scheme to encrypt the file along with the generated key.

Access rights are very important while we are using the cryptree concept but the access rights differ based on files and folder implementation.

The following are the keys that are used for read access on the files and folder structure.

1. Access Key AK: To check whether the user is valid or not
2. Parent Key PK: To find out the information about the parent files
3. File Key FK: To access the files that are in the folder.
4. Reference Key RK: To access the information about the previous folder
5. Metadata Key MK: To access all information about the files and folders. This includes its name, creation date and all the information about the files.

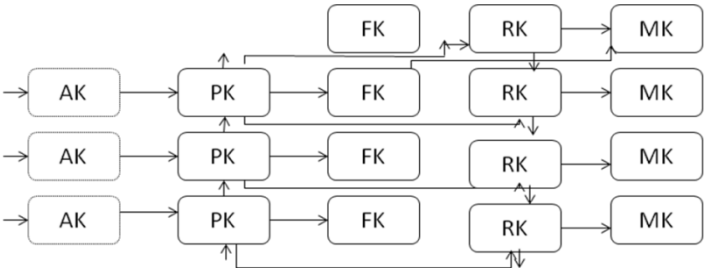

Figure 2. Read access of files and folders

All the details are stored in the form of the log file in the user's database.

1. The following are the keys that are used for write access to the files and folder structure.
2. Access Key AK that can be used to Check whether the user is valid or not
3. Parent Key PK that can be used to find out the information about the parent files
4. Ksign: To provide write access to the files and the folders.
5. Kverify: This key is used to verify the access rights of the user.

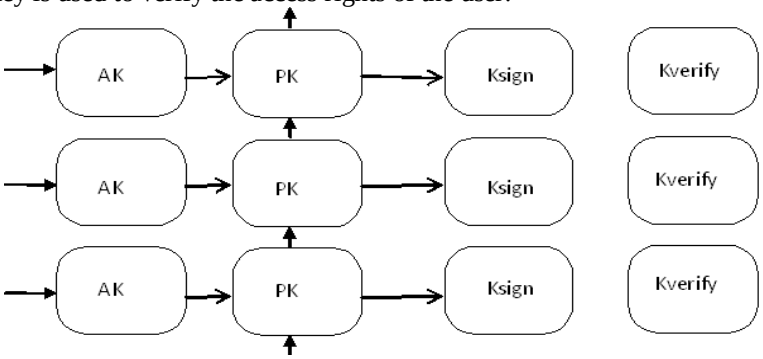

Figure 3 Write access of files and folders

When a user leaves the group or relieves his account or when there is a modification in the group, Access Key (AK) is changed depending on the changes in the group.

V. RESULTS AND DISCUSSION

When the user is authenticated into the groups the system generate the random key. During this process AK access key is generated with a user name and the generated random number. This is later used for give an access permission for a particular user. When a new user enters the cloud system, in this case, an organization, the user is given a random user ID at first. This ID is later updated by the cloud administrator based on the ID of the members present in the group. In the case of occurrence of more of two groups in the cloud, the administrator decides about which group the new user could be placed.

When the file is uploaded into the cloud the user is checked valid access key when this verification is completed the user uploads a file into the cloud. During this time FK is generated using the file name and the time stamp with access permission as read or read/write. Mk is also generated using the file name.

VI. CONCLUSION AND FUTURE WORK

We have proposed a cryptree based group key management service that centralizes keys and key maintaining tasks and manage them to provide security. We encrypt the file using the key generated by MD5 algorithm with AES.

Several keys can be managed by our key management system which is accessible by the user groups. The key management process is highly secure and it is cannot be locked by any other mechanism. This work proposes a new way to provide data security and privacy. Our model introduces a new way of interaction between user groups and cloud service provider. This work also suggests the use of symmetric key cryptography and the use of a MD5 algorithm which are an integral part of my work. Also, there is no need to know the entire master key thereby this method is more secure.

In future, we aim at using Elliptic Curve Cryptography, since it is much more secure and provides strong security with smaller key sizes. Also, we propose to enable key sharing among users so that they can share the files and also enable dynamic members to access keys with the help of the members of the group in a secure way.

REFERENCES

[1] Amar R. Buchade, Rajesh Ingle, 2014, "Key Management for Cloud Data Storage: Methods and Comparisons,", IEEE, Fourth International Conference on Advanced Computing & Communication Technologies (ACCT), pp.263- 270

[2] Dominik Grolimund, Luzius Messier, Stefan Schmid, Roger Wattenhofer, 2006), "Cryptree: A Folder Tree Structure for Cryptographic File Systems", IEEE, 25th IEEE Symposium on Reliable Distributed Systems, pp.189 -198.

[3] Hsing-Chung Chen, Anita Christina, (2014), "A Role-based Key Management Approach in a Hierarchy Scheme", IEEE, Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, pp. 258-264.

[4] Julian Jang-Jaccard, Avnish Manraj, Surya Nepal, (2012), "Portable Key Management Service for Cloud Storage", IEEE, 8th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), pp.147 -156.

[5] Krunal Patel, Navneet Singh, Sendhil Kumar K S, Jaisankar N, (2014), "Data Security and Privacy using Data Partition and Centric key management in Cloud", IEEE, International Conference on Information Communication and Embedded Systems (ICICES), pp.1- 5.

[6] Peter Mell, Timothy Grance,(2011), "NIST definition of cloud computing", National Institute of Standards and Technology.

[7] Roney Thomas, Salim A, (2014), "A Novel Decentralized Group Key Management using Attribute Based Encryption", IEEE, First International Conference on Computational Systems and Communications(ICCSC), pp.358-363.

[8] Sheng Xiao, Weibo Gong, Don Towsley, Qingquan Zhang, Ting Zhu, (2014) "Reliability Analysis for Cryptographic Key Management", IEEE, Communication and Information Systems Security Symposium, pp.

[9] Shikha Sharma, C. Rama Krishna, (2015), "An Efficient Distributed Group Key Management using Hierarchical Approach with Elliptic Curve Cryptography", IEEE, International Conference on Computational Intelligence & Communication Technology, pp.687-693.

[10] Tsung-Chih Hsiao, Tzer-Long Chen, Chih-Sheng Chen, Fu-Sheng Xu, Starlition Tsui, Yu-Fang Chung, Tzer-Shyong Chen, (2013), "Secure Authorization for Controlling Access via Key Management Scheme", IEEE, Region 10 Conference (31194) TENCON, pp.1 – 4.

[11] Xuefeng Liu, Yuqing Zhang, Boyang Wang, Jingbo Yan, (2013), "Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud,", IEEE, IEEE Transactions on Parallel and Distributed Systems, vol.24, no.6, pp.1182 - 1191.

[12] Zhongma Zhu, Rui Jiang (2015), "A Secure Anti-Collusion Data Sharing Scheme for Dynamic Groups in the Cloud," IEEE, IEEE Transactions on Parallel and Distributed Systems, pp. 1- 25.

[13] Zhongma Zhu; Zemin Jiang; Rui Jiang, (2013), "The Attack on Mona: Secure Multi-owner Data Sharing for Dynamic Groups in the Cloud," IEEE, International Conference on Information Science and Cloud Computing Companion (ISCC-C), pp.213- 218.

[14] Jamil T, (2004), "The Rijndael algorithm," Potentials, IEEE, vol.23, no.2, pp.36-38.

[15] Rajesh Ingle, G. Siva Kumar, (2007), "Tunable group key agreement", 32rd IEEE conference on local computer Networks.

