AN AUTHENTICATION BASED SECURE DATA STORAGE IN CLOUD COMPUTING

[1]POTHULA SUJATHA
[1]Department of Computer Science,Pondicherry University,India.
Email:[1]spothula@gmail.com

***Abstract*—Cloud computing has brought a new way that organizations manage their data, due to reduced cost, robustness and ubiquitous nature. However storage and access of sensitive data in cloud computing is a crucial issue. This paper introduces a new cryptographic tree based key management and authentication system to handle this. The tree based key management derivation hierarchy allows only the outsourcing party to access the data block located at a specified node and prevent the access of data block encrypted with child keys. The authentication system checks whether the requested user is valid or not.**

***Keywords*—Tree based, cryptographic, key management, data storage, authentication mechanism, hash function**

I. Introduction

Cloud computing is an internet based infrastructure that provides users to use different kinds of programs without actually installing those programs on their own computers. The term "cloud computing" means the use of different services through the Internet, which are otherwise provided by hardware and software that we want to purchase our own. Cloud computing can be used in business processes and make the employees' administrative tasks easier, grow the business, increase productivity, and generate more revenue. Major advantage of cloud computing applications for business is that, since there is no physical software or hardware necessary other than a Web browser or other interface hardware, the cloud applications can be accessed by anyone, anywhere. So many organizations are using the cloud services for their business.

While large no of companies use cloud storage for storing their private data, which is controlled by un-trusted parties, many issues regarding the privacy and security will occurs. One major issue regarding this is how to secure the confidentiality and privacy of user data, which is shared and used by multiple parties. For providing security the data stored in the cloud are encrypted so that only authorized parties having the decryption key can access them. But here also it is subjected to brute force attack which is a major issue to consider.

The out sourced data generally consists of large number of data blocks so the encryption key management is a big challenge. Many tree based hierarchy schemes have been widely used in group key management. Wang et al. [3] proposed a tree-based cryptographic key management concept for data storage in cloud. This key management structure is almost similar to the traditional key management scheme. In this scheme a single root node is responsible for holding the master key, which can be used to derive the other node keys. With this scheme, a data block stored in the cloud can be updated by a party who holds either the specific decryption key or a node key corresponding to one of its parents. But the problem here is that once a key in the parent node in the tree is known, all the children would be known.

This problem occurs in many tree based key management scheme.

Considering this problem, Miao Zhou, YiMu, WillySusilo, JunYan, LijuDong [1] proposed a Privacy enhanced data outsourcing in the cloud scheme in which a data source protected with a node key in a key management tree can be shared with or managed by another party without compromising the security of the data encrypted with its child node's keys. The encrypted data block associated with a node can be decrypted by multiple decryption keys where one of them is associated with the tree and can be utilized to generate its keys children's keys, while other decryption keys are only used to decrypt the data block stored in the node.

When we generalize this scheme, *n* sub keys are formed which can be used by n different users to access the same data. In this scenario the scheme can be subjected to brute force attack, so that any of the secondary key or even the master key can be known to the attackers. This leads the attacker know about the node data information or even the entire child node data.

Considering this issue this paper introduces an additional authentication mechanism in which an authentication system is provided between the cloud system and the user. The decryption key will be allotted to the user only after authenticated by the cloud owner provided information. So the user who wanted to access the cloud data has to provide the user name and the digital signature allotted by the owner during the registration. The user will be allotted the decryption key to decrypt the data only after the successful verification. After this the user can update the corresponding data in the cloud storage.

II. Related Works

Miao Zhou, YiMu, Willy Susilo, JunYan, LijuDong proposed a Privacy enhanced data outsourcing in the cloud model [1]. This method uses a cryptographic key management tree scheme. A data source protected with a node key in a key management tree can be shared with or managed by another party without compromising the security of the data encrypted with its child nodes' keys. Two decryption keys (d1, d2) will be assigned to a node. d1 is used for the decryption of the database located at the corresponding node and the generation of sub-keys for the child nodes, while d2 can only be used for the decryption of the database at the same node. But has the limitation that many decryption keys can be associated with one encryption key. Thus Brute force attack can be applied to this method so that the attacker can access either the node data or the entire child node data.

Sandeep K Sood proposed a combined approach to ensure data security in cloud computing [2] which is used to provide a way to protect the data, check the integrity and authentication by following the best possible industry mechanisms. The proposed frame work is divided into two phases. First phase deals with process of transmitting and storing data securely into the cloud. Second phase deals with the retrieval of data from cloud and showing the generation of requests for data access, double authentication, verification of digital signature and integrity, thereby providing authorized user with data on passing all security mechanisms. The data on cloud will be stored in encrypted form and the index for searching the data will be also encrypted. MAC code is used to check whether data has been tampered throughout the transmission and this check can be made by the user or owner of data on retrieving the file. By SSL encryption, there is also a key that allows only an authorized person to be able to decode the information. However it includes lots of procedures for verification and access granting, it will take significant amount of time, which will causes the slowing down of access process.

Ulrich Greveler, Benjamin Justus, Dennis Loehr proposed a Privacy Preserving System for Cloud Computing [3], which prevents sensitive information need to be protected not only against external administrators, service providers, but also local administrators . Machine readable rights expressions are used here in order to limit users of the database to a need-to-know basis. This gives a system architecture that allows sufficient and flexible restriction writing. So, local administrators as well as cloud administrators are not able to change the access rules after an application is launched. However the creation of complex machine readable access rights to the decryption keys becomes a challenging problem. When the user-related conditions become sophisticated, the syntax of XML-based rights expressions is complicated and obscure.

Nuno Santos, Krishna P. Gummadi, Rodrigo Rodrigues discussed a trusted cloud computing system [4] which enables IaaS providers to provide a closed box execution environment that guarantees confidential execution of guest virtual machines. It allows users to attest to the IaaS provider and determine whether or not the service is secure before they launch their virtual machines.

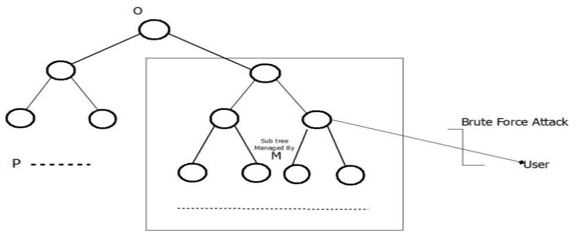
Elisa Bertino, Federica Paci, Rodolfo Ferrini discussed a Privacy-preserving Digital Identity Management for Cloud Computing [5] which provide a privacy enhanced technique to authenticate users and to support flexible access control to services, based on user identity properties and past interaction histories. It Provides a privacy-preserving multi-factor identity attribute verification protocol supporting a matching technique based on dictionaries, ontology mapping techniques, and look-up tables to match cloud service providers and clients vocabularies.

III. Proposed System

Privacy enhanced data outsourcing in the cloud [1] proposes a tree based cryptographic key management scheme for data storage in cloud. The Owner-write-users-read scenario is referred here. It includes 4 parties. They are the cloud provider (P), the data owner (O), Sub-tree data manager (M) and a User (U). The cloud provider is the party which provides the data storage services. The data owner contains the master key and is responsible for key management system. The sub-tree manager holds an authorized node key and that is used to derive all decryption keys for corresponding child nodes. The user will have the permission to use or share a data block at a node managed by M. The Owner O will generate the master key that can be used to derive all other decryption and encryption keys. The entire Sub-tree manager except the leaf will get a master decryption key which can be used to generate all node keys of the sub-tree including the secondary decryption key, which is derived from the master key. User can request a secondary decryption key from M for accessing the encrypted data stored in corresponding node.

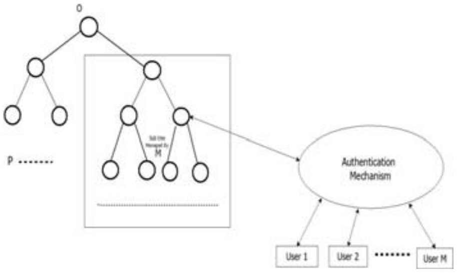
A. *Problem Definition*

In this model, it is expanded to produce *n* decryption keys that are mapped to a single encryption key used to encrypt the corresponding data block. So brute force attack can be applied here and the attacker may get the decryption key. Since n number of decryption keys can be used to derive a single encryption key, an outsider or an attacker can perform the brute force technique to obtain the secondary decryption key or even the master decryption key so that the data stored in the corresponding node or even the data in the all child nodes can be accessed by that attacker. So, in this model we cannot guarantee the safety of the stored data.



"Fig. 1" Problem with the existing model

So in this model we have to incorporate an authentication mechanism to prevent the brute force attack. We will place this Authentication mechanism between the User and Owner/sub-tree manager in order to ensure that whether an authorized user is requesting for a decryption key. The model is shown in figure:



"Fig. 2" Proposed system model

After this authentication, if the user is authorized, the sub-tree manager will allocate a decryption key and owner's digital signature. By using these along with the username only the user can access the cloud to get the data. The authentication mechanism is explained below.

B. *Key Generation*

Consider the stored data contains n blocks, where n lies between $2^{(i-1)} \leq n \leq 2^i$ and *i* denotes the level of the tree. Then generate a complete binary tree from node N_0 to N_{ij}, where *j* denotes the node index. Each node N_{ij} except the root node has the key value K_{ij} with two decryption keys d_{ij1} and d_{ij2}. The two decryption keys are denoted as the key pair (d_{ij1}, d_{ij2}). E_{ij} is the encryption key generated by such a key pair for this node.

The key derivation tree is denoted as T and is T = < N, K >, where N is the node and K is the key value for that node. The tree is rooted at node N_0. Any node in this tree except the leaves can derive its child node. The indices for the derived child will be i(2j-1) for the left and i(2j) for the right, where i=1, 2, 3……. And j=1, 2….. , 2i. The parent of any node except the root is found at index (i-1) (j/2).

The key derivation tree is constructed by using a cryptographic one-way hash function. i.e. H: {0, 1}* →Zq, which can be used to compute the decryption key of child nodes of any node N_{ij}. The key value K_{ij} of node N_{ij} is denoted as $K_{ij} \rightarrow (d_{ij1}, d_{ij2})$, where d_{ij1} denotes the master decryption key and d_{ij2} denotes the secondary decryption key. The secondary decryption key d_{ij2} is computed from the master decryption key d_{ij1} by using the One-way hash function, ie: $d_{ij2} \rightarrow H(d_{ij1})$. The key derivation hierarchy is explained below, where $i \geq 0$, $j \geq 1$.

The root key d_0 for node N_0, we can derive all the key pairs as,

Left child derivation

$d_0 \rightarrow d_{111} \rightarrow d_{112}$

$d_{111} \rightarrow d_{211} \rightarrow d_{212}$

$d_{121} \rightarrow$

$d_{(i-1)(j/2)1} \rightarrow d_{i(j-1)1} \rightarrow d_{i(j-1)2}$

Right child derivation

$d_0 \rightarrow d_{121} \rightarrow d_{122}$

$d_{111} \rightarrow d_{221} \rightarrow d_{222}$

$d_{121} \rightarrow d_{241} \rightarrow d_{242}$

$d_{(i-1)(j/2)1} \rightarrow d_{ij1} \rightarrow d_{ij2}$

The encryption keys are generated from key pairs which contain the master decryption key and the secondary decryption key. i.e. ,

$(d_{111}, d_{112}) \rightarrow e_{11}$ (d121, d122) $\rightarrow e_{12}$

$(d_{211}, d_{212}) \rightarrow e_{21}$ (d221, d222) $\rightarrow e_{22}$

.

$(d_{i(j-1)1}, d_{i(j-1)2}) \rightarrow e_{i(j-1)}$ $(d_{ij1}, d_{ij2}) \rightarrow e_i$

.

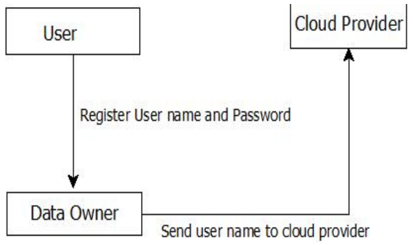
The original data owner *O* conducts some computations to derive child keys. For a key pair (d_{ij1}, d_{ij2}) of node (i, j), its child on the left can be calculated as $(d_{(i+1)(2j-1)1}, d_{(i+1)(2j-1)2}) = (H(d_{ij1}||(2j-1)), H(H(d_{ij1}|| (2j-1))))$ and its right child can be calculated as

$$d_{(i+1)(2j)1}, d_{(i+1)(2j)2}) = (H(d_{ij1}||2j), H(H(d_{ij1}|| 2j)))$$

Other sub-keys can generate accordingly. In this way, the whole key derivation tree can be constructed.

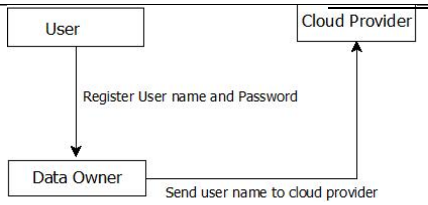
C. *Authentication Mechanism*

In the Authentication, the user who all wants to access the data through cloud need to register with the data owner. During registration, the Data owner asks for a user name, password and a security question. After registration, the Owner sends the registered user name to cloud provider and the cloud provider stores that in their data base.


User
Cloud Provider
Register User name and Password
Data Owner
Send user name to cloud provider

"Fig. 3" Registration procedure for the user

The data owner allots the user name and password for the sub-tree managers and it sends the user name to the cloud provider for checking during the access request.

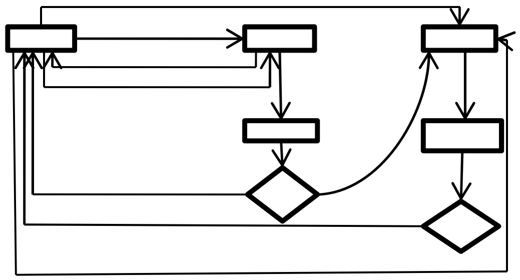

User
Cloud Provider
Register User name and Password
Data Owner
Send user name to cloud provider

"Fig. 4" Registration procedure for the Sub-tree Manager

After that if a user want to access the data, first it will send a request to access the decryption key to the Owner/Sub-tree manager along with their user name. Then the Owner/Sub-tree manager will ask the password and the security question provided during the time of registration. So the User wants to send the user name and answer of the security question. The Owner/Sub-tree manager will check and verify the password and security question by comparing with their data base. If both match, it will send the decryption key and Owner's digital signature back to the User. Also it will send the user identity and owner's digital signature to the cloud provider. Otherwise it will discard the request from the user. After getting the decryption key and owner digital signature, the user will send a request to access data along with digital signature and user name to the cloud provider.

The cloud provider will verify the digital signature and user name which are already stored in its database. If both match, the cloud provider will send the encrypted data blocks to the user. So by getting the encrypted data blocks, the user will decrypt this by using the decryption key which is obtained from the Owner. Then after performing its work, the user can update the data in the cloud.

Since after the verification by the data owner it will send the owner's digital signature which is used for the second verification by the cloud provider, even if an attacker got the decryption key through brute f force attack, it can't access the data blocks in the cloud.



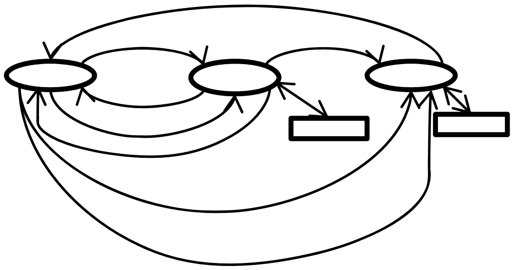
"Fig. 5" Proposed Model

It is because the attacker won't be able to get the owner's digital signature for that instance. So the brute force attack will not occur. Not only that in this during the request to access the decryption key, the Owner will ask for the security question which is given during the registration process. So in cases such as any user looses or by mistake reveals his user identity and password to any unauthorized person, that data can't be accessed by that unauthorized person. Because that answer is known to the authorized user only. Moreover, attacker must know the master key to decrypt the encrypted data received from the cloud. Here one more authentication about the user name is performed by the cloud provider when they get the request to access the data, which will give this system an additional security.

IV. Data Request and Access Mechanism

The data access procedure for the proposed system is given as

"Fig. 6" Data access between cloud providers and sub-trees Manager



"Fig. 7" Data access between user and Cloud provider

V. Conclusion and Future Work

Security and privacy of outsourced data is one of the major challenges in the cloud computing. To overcome this challenge cryptographic tree based key management system has been used. The cryptographic tree based key management derivation hierarchy allows only the outsourcing party to access the data block located at a specified node and prevent the access of data block encrypted with child keys. In order to provide more security to this existing system, the proposed technique introduces an additional authentication mechanism between the cloud provider and the user. The authentication system checks whether the requested user is valid or not. By combining this additional authentication mechanism in the system we will get a high efficiency data outsourcing model. In future the proposed technique will be implemented to verify its high efficiency and security in comparison to other existing techniques.

References

[1] Miao Zhou , YiMu, Willy Susilo ,JunYan and Liju Dong, "Privacy enhanced data outsourcing in the cloud", *Journal of Network and Computer Applications*, vol.35 no.4, pp.1367–1373, 2012.

[2] Sandeep K. Sood, "A combined approach to ensure data security in cloud computing", *Journal of Network and Computer Applications,* vol. 35 no. 6, pp. 1831–1838, 2012.

[3] Wang W, Li Z, Owens R and Bhargava B, "Secure and efficient access to outsourced Data", *Proc. of the ACM workshop on cloud* computing *security*, New York, pp.55–66, 2009.

[4] Ulrich Greveler, Benjamin Justus and Dennis Loehr, "A Privacy Preserving System for Cloud Computing" *Eleventh IEEE international conference on computer and information technology*, pp. 648–653, 2011.

[5] Nuno Santos, Krishna P. Gummadi and Rodrigo Rodrigues, "Towards Trusted Cloud Computing", *Proc. Hot Cloud*, 2010.

[6] Elisa Bertino, Federica Paci and Rodolfo Ferrini, "Privacy-preserving Digital Identity Management for Cloud Computing" *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, pp.1-72009.

[7] Sabrina De Capitani di Vimercati, Sara Foresti, Sushil Jajodia, Stefano Paraboschi and Pierangela Samarati, "A Data Outsourcing Architecture Combining Cryptography and Access Control", *ACM Workshop on Computer Security Architecture*, Fairfax,Virginia, USA, November 2007.

[8] C. Blundo and S. Cimato and S. De Capitani di Vimercati, A. De Santis andS. Foresti and S. Paraboschi and P.Samarati, "Efficient Key Management for Enforcing Access Control in Outsourced Scenarios", *Springer Boston, Proc. of IFIP"* pp. 364–75, Boston: Springer; 2009.

[9] Xiang Zou, Bing Chen and Bo Jin," Cloud-Based Identity Attribute Service with Privacy Protection in Cyberspace", *Procedia Engineering,* Vol. 29, pp.1160 – 1164, 2012.

[10] Loganayagi.B and S.Sujatha, "Enhanced Cloud Security by Combining Virtualization and Policy Monitoring Techniques", *Procedia Engineering,* vol. 30, pp. 654 – 661, 2012.

[11] Fa-Chang Cheng and Wen-Hsing Lai, "The Impact of Cloud Computing Technology on Legal Infrastructure within Internet—Focusing on the Protection of Information Privacy",*Procedia Engineering,* vol. 29, pp.241 – 251, 2012.

[12] Dimitrios Zissis and Dimitrios Lekkas, "Addressing cloud computing security issues", *Future Generation Computer Systems,* vol. 28 pp.583–592; Elsevier, 2012.

[13] Chunming Rong, Son T. Nguyen and Martin Gilje Jaatun, "Beyond lightning: A survey on security challenges in cloud computing", *Computers and Electrical Engineering, vol.* 39 pp. 47–54, Elsevier, 2012.

[14] Nir Kshetri, "Privacy and security issues in cloud computing: The role of institutions and institutional evolution", *Telecommunication Policy*, vol. 37, no.4-5, pp. 372-386, Elsevier; 2013.

[15] Cloud computing world home page http://Cloudcomputingworld.org.