An efficient Framework for Multi-Authority of Data Storage and Access Control

G. Kanaka Durga
Student, II/II M. Tech

D. Srikar
Asst. Professor, GVVIT, A. P India.

Abstract—In cloud services there are many data owners to store their data in cloud. There increases more scalability and more traffic between the users. The data which is shared between the users is at more risk, because of more scalability of users are access the files. We introduced a novel method of having multi authority on cloud storage by multiple data owners and multiple users. We used attribute based encryption to authenticate the user and for the security purpose we introduced block encryption process to provide more security to message and the exchange of the data.

Index Terms—Access control, multi-authority, CP-ABE, attribute revocation, cloud storage.

I. INTRODUCTION

The cloud storage is the vital scheme in cloud computing. Which provide services to the Cloud storage is a vital service of cloud computing, It will provide service like exact data accessing from the database and control on the data when we give permissions to the multi users to access the same data. We have to use this scheme for greater access of data and to provide security to the data. Because the owner unable to trust any server. But the owner can trust a server whenever it uses data access control scheme. Because it will give direct access to the owner that means the owner can access his data directly. But the CP-ABE scheme should maintain the management and key list also. The permission to the owner gives when only the authentication requirements satisfies. The owner of the data can encrypt the data by using the technique what are defined in the scheme. The encryption permission only gives to the owner only. The user can decrypt the data he doesn't have any permission to edit, encrypt the data. This system will give permission to the owners those who are already having registered identification in the university or have identification in government.

In this CP-ABE there are two types of systems, first one is single authority means that all the attributes and keys are handled by the single person only. And second one is multi authority in this all the attributes and keys are handled by the many persons. The persons may be owners and administrators of the server. By using this type of data access control we can provide many authorized persons to the same cloud data storage systems. Through using this type of data access controller we can provide onward safety and as same as toward the back safety also.

Scheme	Authority	Revocation Message	Backward Security	Forward Security	Revocation Enforcer	CT Updater
[11]	Single	$O(n_{non,x}\log\frac{n_u}{n_{non,x}})$	Yes	Yes	Server*	Server*
[13]	Multiple	$O(n_{c,x}\cdot n_{non,x})$	Yes	No	Owner	Owner
[14]	Multiple	$O(n_{c,aid}+n_{non,x})$	Yes	Yes	AA	Server†
Our	Multiple	$O(n_{non,x})$	Yes	Yes	AA	Server†

∗: The server is fully trusted; †: The server is semi-trusted.
$|p|$ is the size of element in the groups with the prime order p; n_u denotes the number of users in the system; $n_{non,x}$ denotes the number of non-revoked users who hold the revoked attribute x and $n_{c,x}$ is the number of ciphertexts which contain the revoked attribute x; $n_{c,aid}$ denotes the total number of attributes belongs to the AA_{aid} in all the ciphertexts.

We just modified the framework of the system and make it very comfortable to the cloud storage system. In this

new design the owner of the data does not involve in key generation. And in this the key of the end user don not mach with the owner's key. In this scheme user has to hold a single password itself no need of holding so many passwords. It is responsible for generating public key and also as well as to generate secret key for each user.

II. SYSTEM AND SECURITY MODELS

2.1 System Model

As discussed in the above we support multi authority for data access. In this there are five types of entities in the system. They are certificate authority (CA), attribute authorities (AAs), data owners (owners), the cloud server (server) and data consumers (users).

The certificate authority is a worldwide trusted official document right in the scheme. It will give registration to the all users and attribute authorities. It will give unique identity to each user and also produce public key for each user. The certificate authority is never involve in the data management and also not involved in the secret key generation.

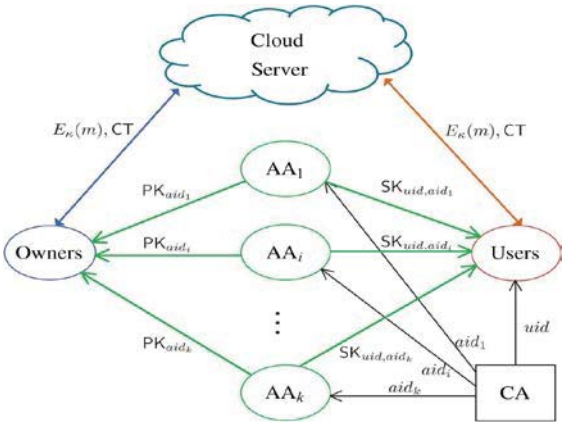

Fig. 1. System model of data access control in multi-authority cloud storage.

The attribute authorities are responsible for entitling and revoking user's attributes according to their role or identity in its domain. In this scheme every attribute having relationship with a single AA, but a single AA can manage number of attributes. So here one to many relation will be worked out. It has total information about the structure and meanings of each attribute. It is responsible for generating public key and also as well as to generate secret key for each user.

The major task of an owner is to split up total data into small pieces based upon his logic representation. He has to encrypt the data by following several policies. He is the person to give authorities to the multiple persons. He will decide all the permissions among the data. The owner only sends his data to cloud server in an encrypted format. Then the cryptography will give data access control over that data. If the user has permission then only he can decrypt that data and then he can use it.

The encryption permission only gives to the owner only. The user can decrypt the data he doesn't have any permission to edit, encrypt the data. This system will give permission to the owners those who are already having registered identification in the university or have identification in government. The Cipher text-Policy Attribute-based Encryption (CP-ABE) is the best technique to provide data access control over the network.

2.2 OUR DATA ACCESS CONTROL SCHEME

In this scheme, the overview of the problems and techniques will be described first. Then the designing will be started which consists of five modules are, System Initialization, Key Generation, Data Encryption, Data Decryption and Attribute Revocation.

2.3 OVERVIEW

First of all the designer should know all the information about multi authority protocol. To design data access control for multi authority cloud storage systems the main challenges are two. They are Security issue and Revocation Issue.

We propose a new revocable multi-authority CP-ABE protocol based on the single-authority CP-ABE proposed by Lewko and Waters in. That is we extend it to multi-authority scenario and make it revocable. We apply the

techniques in Chase's multi-authority CP-ABE protocol to tie together the secret keys generated by different authorities for the same user and prevent the collusion attack. Specifically, we separate the functionality of the authority into a global certificate authority (CA) and multiple attribute authorities (AAs). The CA sets up the system and accepts the registration of users and AAs in the system. It assigns a global user identity uid to each user and a global authority identity aid to each attribute au-thority in the system. Because the uid is globally unique in the system, secret keys issued by different AAs for the same uid can be tied together for decryption. Also, because each AA is associated with an aid, every attribute is distinguish-able even though some AAs may issue the same attribute.

To deal with the security issue instead of using the system unique public key (generated by the unique master key) to encrypt data, our scheme requires all attribute authorities to generate their own public keys and uses them to encrypt data together with the global public parameters. This prevent the certificate authority in our scheme from decrypting the ciphertexts.

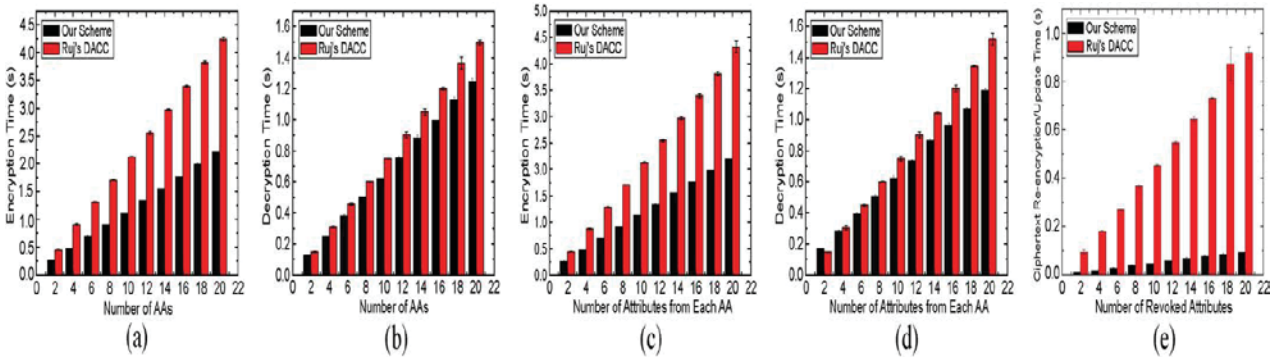

Fig. 3. Comparison of Computation Time.

(a) Encryption. (b) Decryption. (c) Encryption.d) Decryption. (e) Re-encryption.

III. RELATED WORK

3.1 Overview

This chapter presents the problem definition and solution concerns used in conducting this project work and provide a justification for each step taken. It involves the problem identification, statement of problem and proposed methods.

3.2 Problem Identification

In traditional or existing system, the data encryption is based on the symmetric key and the private key of the data owners. In the existing system, data owners shares the key to use the file by the users. In this work previously traditional attribute based encryption is used to provide authentication for each and every user. The data authentication and verification is based on the symmetric cryptography only. The data exchange becomes very risky process. The data owner has to share the secret key for every user. This increases the processing of the user and more time complexity. Data owner has to authenticate the every use request for verification.

3.3 PROBLEM STATEMENT

In this project we are proposing an efficient revocable data access mechanism for multi authority in cloud data access. We are improving the current attribute based encryption for secure outsourced data between data owners. Certification authority generates and distributes the key to all the authenticated users or data owners. Our proposed model more concentration on group security with key management protocol and authentication. This approach improves the performance than previous models with simple and novel architecture

3.4 Proposed Technique

In our work we introduced a novel architecture to provide more security for the data owners and their private data on multi authority of the file. There are some sequence of steps to process this method as shown below:

Architecture

3.4.1 Setup: The data owner has to register in cloud service provider. The cloud service provider provides a master key for the data owner by taking the input of data owner identity.

$$D_m = MK(ID);$$

For the end user the data owner has to provide the security for accessing the data. But the verification process will execute by the cloud service provider. Every user have their unique ID.

$$U_k = S_k(U_{id});$$

3.4.2 Authentication

For this process we used key attribute based encryption. In this by using the master key cloud service provider issues a hash code using sha1. This code will be the key to every data owner authentication when manipulating the data and the user verification.

$U_v = Hash(DW_r)$ Where r is random number generated by the data owner.

3.4.3 Multi authority

In this process the data owner has to issue the authority to access the data by another data owner. The original or root data owner issues the privilege to another data owners to access his data on demand. This process is based on the authentication process.

3.4.4 Attribute based encryption

In cipher text-policy attribute-based encryption (CP-ABE) a user's private-key is associated with a set of attributes and a cipher text specifies an access policy over a defined universe of attributes within the system. A user will be able to decrypt a cipher text, if and only if his attributes satisfy the policy of the respective cipher text. Policies may be defined over attributes using conjunctions, disjunctions and (k, n)-threshold gates, i.e., k out of n attributes have to be present (there may also be non-monotone access policies with additional negations and meanwhile there are also constructions for policies defined as arbitrary circuits). For instance, let us assume that the universe of attributes is defined to be {A, B, C, D} and user 1 receives a key to attributes {A, B} and user 2 to attribute {D}. If a cipher text is encrypted with respect to the policy$(A \vee C) \vee D$, then user 2 will be able to decrypt, while user 1 will not be able to decrypt.

3.4.5 Message Encryption and Decryption

We encrypt the data in the format of blocks. Initially we divide the message into blocks then we encrypt on decrypt in different locations. Every Block is encrypted and decrypted as shown below:

1. Input: ASCII values of input message or file

Example: if character in message is 'A' the ASCII value is 65

2. Calculate length of input message /file
Example: m=512
3. Enter block size
Example: n= 128
4. Initialize reserved bytes
Res= 16 bytes
5. Calculate how many zeroes we have to append to divide the blocks in the size of 'n' for length of File/message.

P=m mod n
P=512 mod 128=0
Q=n-(P + Res)
Q=128-(0+ (16*8))
Q=0
6. If Q > 0
Append Q number of zeroes to input message or file.
Else
Append 'n + Q' zeroes to input message or file.
7. Calculate length of file/message after appending of zeroes.
After appending of zeroes
Our example: l=640
8. Calculate number of blocks
Count=640/128 = 5
9. for 1 to count
S = reverse [for 1 to count [((A⊕B) ∨ (A ∧ B))]]
For example take one character 'A'
A=1, B= 65
Calculate the above S and reverse that result.
Calculate each and every block signature up-to count.
After encryption and decryption the data owner shares the information in cloud service provider and gives privileges to neighbouring data owners and on demand users also view the content of the file or data.

3.5 Architecture diagram: In this architecture we aimed to restrict the access privileges of the data owners to manipulate the files. The goal is to verify not only the authentication; we verify the
Data and the user authorization.

3.5.1 Data owner: The data owner uploads the file using the block based encryption and signature. He can manipulate his file and grant access privileges to neighbour data owners to manipulate the file.

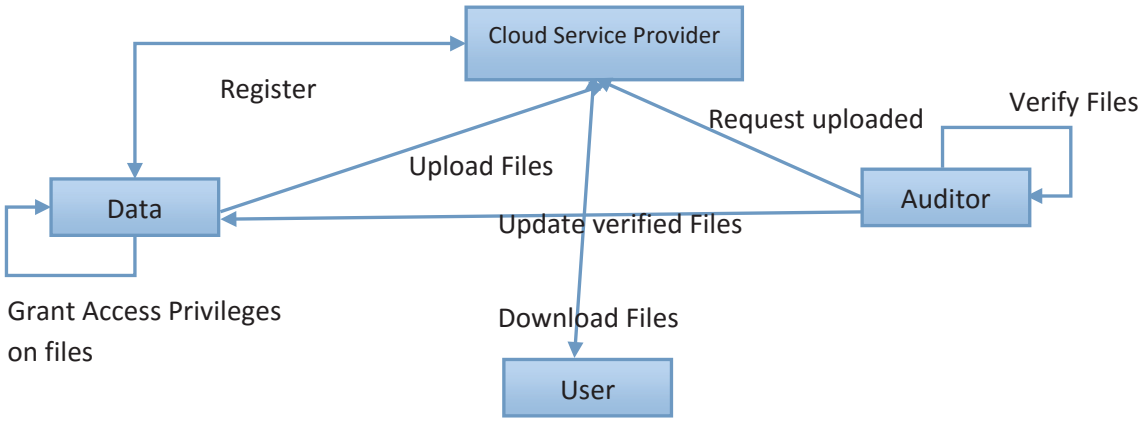

Fig 4: Architecture diagram, relations among entities i.e., Data, User, Auditor

3.5.2 Cloud Service Provider: It stores the data owner's files and validates the users at the time of log in and uploading files. It verifies all types of users who are accessing the files in its database.
3.5.3 Users: Users don't have manipulation rights; they can only download the files.

3.5.4 Auditors: Auditors view the uploaded files in the CSP. Auditor verifies the files and updates the status to the data owner using SMTP Protocol.

IV. CONCLUSIONS

In this project we proposed multi authority scheme, which supports efficient attribute recall. By using this we established multi authority cloud storage systems. By using scheme we can provide security to the data. So by using this we can use this in social network and at any remote storage systems to store and access our data efficiently.

REFERENCES

[1] P. Mell and T. Grance, ‘‘The NIST Definition of Cloud Computing,’’ National Institute of Standards and Technology, Gaithersburg, MD, USA, Tech. Rep., 2009.

[2] J. Bethencourt, A. Sahai, and B. Waters, ‘‘Cipher text-Policy Attribute-Based Encryption,’’ in Proc. IEEE Symp. Security and privacy (S&P’07), 2007, pp. 321-334.

[3] B. Waters, ‘‘Cipher text-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization,’’ in Proc. 4th Int’l Conf. Practice and Theory in Public Key Cryptography (PKC’11), 2011,pp.53-70.

[4] V. Goyal, A. Jain, O. Pandey, and A. Sahai, ‘‘Bounded Cipher text Policy Attribute Based Encryption,’’ in Proc. 35th Int’l Colloquium on Automata, Languages, and Programming (ICALP’08), 2008, pp. 579-591.

[5] A.B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B.Waters, ‘‘Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption,’’ in Proc. Advances in Cryptology-EUROCRYPT’10, 2010, pp.62-91.

[6] M. Chase, ‘‘Multi-Authority Attribute Based Encryption,’’ in Proc. 4th Theory of Cryptography Conf. Theory of Cryptography (TCC’07), 2007, pp. 515-534.

[7] M. Chase and S.S.M. Chow, ‘‘Improving Privacy and Security in Multi-Authority Attribute-Based Encryption,’’ in Proc. 16th ACM Conf. Computer and Comm. Security (CCS’09), 2009, pp. 121-130.

[8] A.B. Lewko and B. Waters, ‘‘Decentralizing Attribute-Based Encryption,’’ in Proc. Advances in Cryptology-EUROCRYPT’11, 2011, pp. 568-588.

[9] S. Yu, C. Wang, K. Ren, and W. Lou, ‘‘Attribute Based Data Sharing with Attribute Revocation,’’ in Proc. 5th ACM Symp. Information, Computer and Comm. Security (ASIA CCS’10), 2010, pp. 261-270.

[10] M. Li, S. Yu, Y. Zheng, K. Ren, and. Lou, ‘‘Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption,’’ IEEE Trans. Parallel Distributed Systems, vol. 24, no. 1, pp. 131-143, Jan. 2013.

[11] J. Hur and D.K. Noh, ‘‘Attribute-Based Access Control with Efficient Revocation in Data outsourcing Systems,’’ IEEE Trans. Parallel Distributed Systems, vol. 22, no. 7, pp. 1214-1221, July 2011.

[12] S. Jahid, P. Mittal, and N. Borisov, ‘‘Easier: Encryption-Based Access Control in Social Networks with Efficient Rev Ocation,’’ in Proc. 6th ACM Symp. Information, Computer and Comm. Security (ASIACCS’11),2011,pp.11-415.

[13] S. Ruj, A. Nayak, and I. Stojmenovic, ‘‘DACC: Distributed Access Control in Clouds,’’ in Proc. 10th IEEE Int’l Conf. TrustCom, 2011, pp. 91-98.

[14] K. Yang and X. Jia, ‘‘Attribute-Based Access Control forMulti-Authority Systems in Cloud Storage,’’ in Proc. 32th IEEE Int’l Conf. Distributed Computing Systems (ICDCS’12), 2012, pp. 1-10.

[15] D. Boneh and M.K. Franklin, ‘‘Identity-Based Encryption from the Weil Pairing,’’ in Proc. 21st Ann. Int’l Cryptology