Hierarchical Attribute based Encryption in Cloud Environment

Sapana Kapadnis
Computer Department,
MET BKC Adgaon, Nashik,Savitribai Phule Pune University, Maharashtra India.

ABSTRACT
In the IT industry cloud computing is considered as one of the most important paradigms. In cloud computing we have to store our data at cloud providers place. While doing data storage and management cloud security and privacy on outsource data increasing these days. Attribute Based Encryption (ABE) techniques came into existence for secure access control like Ciphertext-Policy Attribute-Based Encryption (CP-ABE) and Key-Policy Attribute-Based Encryption (KP-ABE). These techniques suffer from problems in implementing flexible and scalable access control mechanisms. This paper provides Hierarchical Attribute-Based Solution in cloud environment. On outsourced data it provides both efficient and flexible access control in cloud computing. In addition it provides multiple value assignment for expiration time to deal with features like user revocation.

Keywords
Access Control, Cloud Computing, Security, Privacy.

1. INTRODUCTION
Cloud computing is a model which provides anything 'as a service'. It allows user to access large amount of data. Large group of servers are networked which are responsible for data sharing task. It provides pay per use service so due to that cost is reduced and also scalability and flexibility is achieved Services provided by cloud are SaaS (Software as a Service) PaaS (Platform as a Service), IaaS (Infrastructure as a Service). In SaaS software is placed centrally and given for use to the user. Example of SaaS is gmail which application service[1]. Cloud provides these services over internet. Cloud have different deployment models as public, private, hybrid, and community model. Access allowed by each model is different. In PaaS computing platform is provided as service. Google app engine comes under the PaaS. IaaS provides network, storage, servers to the user. Amazon.com [4] provides infrastructure as a service. Services provided by cloud have to face many problems like as data stored on cloud control on data is lost. Infrastructure management and data management is provided by cloud service provider, this is the risky part of cloud computing. If it is public cloud management interfaces of customers are accessible from internet. Data deleted at cloud may not get deleted. Security problem in cloud computing is become serious these days. Security in cloud computing is major concern as whole data is transferred through internet. For storage and business operations user have to give their data to cloud service provider, which are not totally trust worthy. So for protecting data mechanisms are provided like access control, data authentication and authorization even auditing also comes under this category. To provide data confidentiality encryption is provided to data. Encryption provides such functionality that data on cloud may not get compromised. For achieving flexible access control many schemes came into existence. The problem with these schemes is that they work when service provider and service consumer are in the same domain. To overcome this problem in an attribute based encryption schemes of different kinds are presented. However, they lack in scalability when attributes have multiple levels. Wan et al. presented a more secure and scalable scheme known as hierarchical attribute set based encryption for access control [1]. It is extension of ASBE algorithm with a hierarchical structure. HASBE [1] provides support for hierarchical user grant also file creation, file deletion and user revocation.

The rest of the paper is structured as follows. Section II reviews relevant literature. Section III provides details about system model. Section IV describes actual working of system. Section V concludes the paper.

2. LITERATURE REVIEW
A. Traditional symmetric/public key cryptosystem: In traditional symmetric cryptosystem [6], key is shred between sender and receiver as key for encryption and decryption . If sender wants to encrypt t he file, he choose one encryption key and send it receiver and encrypt the file. The ciphertext generated by sender is stored at cloud. At receivers end by using same encryption key. In public key cryptosystem [6] for every user one public key and private key is assigned. Data is encrypted using public key and can be only decrypted using corresponding private key.

B. Broadcast Encryption: In broadcast encryption (BE)[6] for some subset S of users broadcaster encrypts a message. Users belongs subset S can only decrypt a message using their private key. In BE there are two parties broadcaster and multiple users. Broadcaster generate secret key for all the users and encrypts the message using this secret key. It uses "one-to-many encryption".

C. Hierarchical Identity Based Encryption: It is same as public key encryption where public key belong to unique user and private key computed by trusted third party (TTP). Only one trusted third party is is present for generation of private key. So reduce workload of this TPP hierarchical identity based encryption (HIBE)[6] is introduced which follows two-level scheme. In this scheme root TPP generates private key for domain-level i.e. first level and domain-level generates private keys for users which comes under their domain.

D. Attribute-Based Encryption: ABE was first proposed by Sahai and Waters [4]. Most of data stored on cloud in encrypted form, but it may be the case that data may get compromised. To provide security to data it must encrypt at fine-grained level. In this scheme key of user and ciphertext provided some labeled attribute set decryption can be possible only if attribute of users key and ciphertext are identical [5].

Key-Policy Attribute Based Encryption (KP-ABE) Sensitive data is shared and stored by third party on Internet, it necessary to encrypt that data. In KP-ABE ciphertext are

labeled with sets of attributes and private keys are associated with access structure that control which ciphertext a user is able to decrypt. A user is able to decrypt a ciphertext if the attribute the attribute associated with a ciphertext satisfy the key's access structure.

Ciphertext Attribute Based Encryption (CP-ABE): User's private key will be associated with an arbitrary number of attributes expressed as strings. When party encrypts a message it specifies access structure over attributes. If user's attributes pass through the ciphertext's access structure then and then only user can decrypt a ciphertext[6].

E. Access control solution for cloud computing : In traditional method data stored at third parties place is stored in encrypted form on server, and decryption keys are given to authorized users. Its drawback is effective key management. Data owner must be online all time for key encryption and decryption.

So, to overcome drawback of all these scheme hierarchical attribute based solution is introduced, it provides flexible and scalable access control.

3. SYSTEM MODEL

Tree Access structure: R. Bobba and H. Khurana[8] has introduced concept of tree access structure in which nonleaf nodes are threshold gates and leaf nodes are attributes. In fig. 1 'AND' and 'OR' gates are used tree only demands HOD in computer department or MET BKC of level value larger than 5. Access structure and attribute sets are compared if attribute set satisfy tree access structure then access is provided to user belongs to that access tree structure.

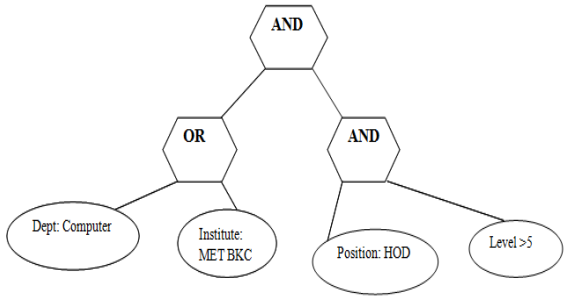

Figure.1 Tree Access Structure

It is based on the work of scientist Zhiguo Wan, Jun'e Liu, and Robert H. Deng. Fine-grained access control is achieved as data owner can define expressive policy for file access. This resulted in an efficient system response time as well as increased performance of the system. For security purpose, it 3 keys are provided private key, public key and master key. scalability is achieved by distributing the workload within hierarchical structure. Another feature provided is User Revocation that allows expiration of user's key to be updated after the duration of key is near to expiration. System model consists of 5 types of components as in Fig. 1 data owner, data consumer, cloud service provider, domain authorities, and a trusted authority.

1. All services related to data storage are managed by cloud service provider.
2. Data owner encrypts file using ciphertext, tree access structure and random data encryption key they store their encrypted files on cloud.
3. Data consumer download encrypted files from cloud as per interest and decrypt them.
4. For authorization and management of data owner and consumer domain authority is present.
5. Management of domain authorities is done by trusted authority it is main root level authority.

Here, each top-level domain authority represents top-level organization and lower-level domain authority represents lower-level organization. Data owner and data consumer may belong to group of employee. It is not necessary that data owner and consumer will be always online. Cloud service provider, top-level domain authority and trusted authority are always online. Data owner can create, delete and update data file while they have all access rights while data consumer can access file in read only mode.

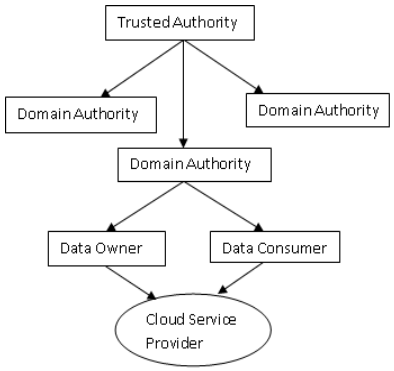

Figure.2 Hierarchical Structure of Users

4. HASBE SCHEME

Key Structure: In HASBE recursive set based key structure is used where each element is either a set or an element related to attribute. *Depth* of the key structure decides the level of recursive set. Member set at depth 1 can either sets or attribute elements but members at depth 2 may only attributes elements. Consider the example where {Dept: Computer, Institute: MET BKC, Position: Lecturer, Level: 3}, {Position: HOD, Level: 6} is key structure of depth 2. It represents attributes related to one person who is tester at level 3 and manager at level 6 as shown in figure 2. Key structure defines the unique labels to the sets. Suppose we have recursive attribute set as $A=\{A_0, A_1, \ldots\ldots\ldots, A_m\}$ where $A_i=\{a_{i,1}, a_{i,2}, \ldots\ldots\ldots, a_{i,n}\}$ in which A_i and n_i are the number of attributes in A_i. And further we can use subset of A[1].

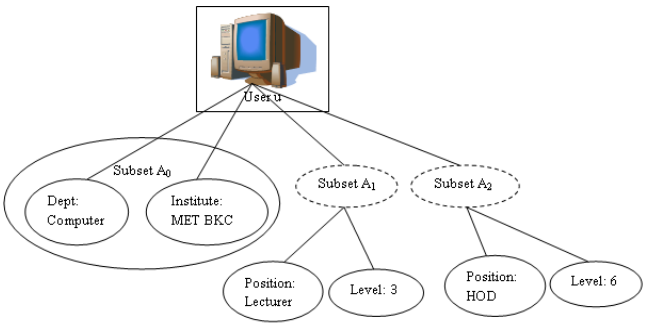

Figure.3 Key Structure Example

HASBE is extension of ASBE to handle hierarchical structure. Main steps in HASBE: system setup, Top-Level Domain authority Grant, New Domain/User Authority, New File Creation, User Revocation, File Access and File Deletion.

- **System Setup:** Trusted authority is responsible for generation of public key and master key. Public key made public to other parties and master key kept secret.
- **Top-Level Domain Authority Grant:** A domain authority has one unique ID and set of attribute. By using master key, public key and attribute set master key for top-level domain authority is generated as shown in Fig. 4.
- **New Domain/User Authority:** Master key of top-level domain authority, some attributes from attribute set and unique id of user creates secret key for new user.
- **New File Creation:** Before storing data on cloud first it should be encrypted and then store that data file on cloud. Each file is encrypted with some data encryption key i.e DEK. Unique ID is chosen for data file then randomly data encryption is selected. By using this data encryption key file is encrypted. Public key, message and tree access structure are encrypted together.
- **User Revocation:** In process of revocation user unable to access any data files after revocation. For this purpose re-encryption is done to revoked user without affecting privileges of other user's.

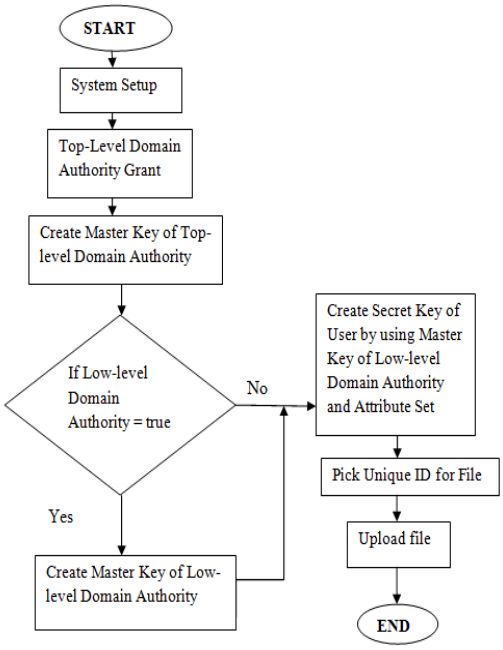

Figure. 4 Flowchart for uploading a file at data owner's side

- **Key Update:** New attribute called expiration_time is introduced in this scheme by using value of this attribute existing key can be updated.
- **Data Re-encryption:** When data owner wants to re-encrypt a data file value of expiration_time is changed and according to that value ciphertext is calculated. By using this new ciphertext value again data file is encrypted.
- **File Access:** As request from user for data access is received cloud service provider send corresponding ciphertext to the user. User decrypt that data file by first decryption using secret key and ciphertext and then by using DEK. Verification of key access structure is done first associated with that particular ciphertext as shown in figure 4. If that attribute set does not match with access structure then access is denied to user else access is granted.

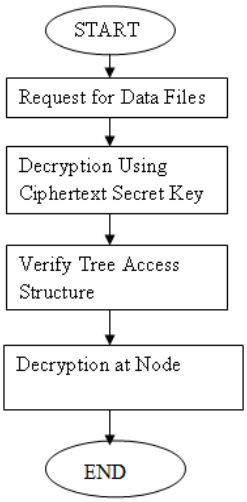

Figure.5 Flowchart for file downloads at data consumer's side.

- **File Deletion:** Data files can be deleted on the request of the data owner. Data owner sends some unique ID and its signature. After successful verification data owner can delete data file from cloud.

This scheme can be used for organization which contains department and various users under that department. As in case of cloud most of the users have their virtual machine and data flows from virtual machine to hypervisor and hypervisor to disk. When data flows from virtual machine to hypervisor at that time we can provide this attribute based encryption, where public and master is generated at organization level then by using this master key and attribute set further generation of master key is done. At user level also master key generated by using master key of department.

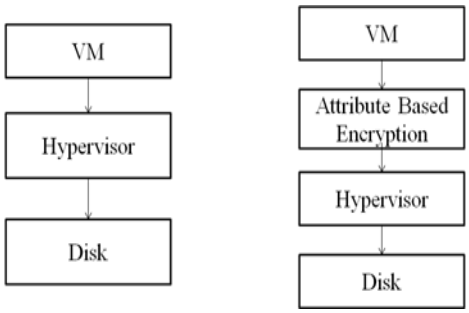

Fig. 6 Encryption at Hypervisor Layer

And same way as in Fig.5 we can upload file and as shown in Fig.6 we can download file.

Advantages of HASBE scheme

1. Scalability: In HASBE scheme because of hierarchical structure of user's great scalability is achieved. Workload for Key distribution and generation of trusted authority is distributed among top-level domain authority and low-level domain authority.

2. Flexibility: User attributes are organized in recursive set structure and allow user to impose dynamic constraints. It supports compound attributes and multiple numerical assignments.
3. Fine-grained access control: Data owner can define and change access policies means access policies are flexible.
4. Efficient User Revocation: As new attribute called expiration_time is introduced because of that user revocation became easy.
5. Expressiveness: It is somewhat similar to Role-Based Access Control so easy to apply HASBE scheme.

5. CONCLUSION

In this paper different encryption schemes used in cloud like ABE, KP-ABE, CP-ABE, and HASBE , where main focus is on access control policies. Random access of data on cloud is restricted as per user and privileges. In KPABE scheme, data is associated with attributes and attribute policies are associated with keys. In CP-ABE schemes, attribute policies are associated with data and attributes are associated with keys and only keys which are associated with attributes can only decrypt the data. HASBE provides the scalable and flexible access control and also, maintains the hierarchy of users for efficient access of data. Also by using this scheme when data flows from virtual machine to hypervisor at that time we can provide this attribute based encryption, where public and master is generated at organization level then by using this master key and attribute set further generation of master key is done.

6. REFERENCES

[1] Zhiguo Wan, Jun'e Liu, and Robert Deng,"HASBE:A Hierarchical Attribute-Based Solutionfor Flexible andScalable Access Controlin Cloud Computing", IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 7, NO. 2, APRIL 2012.

[2] R. Buyya, C. ShinYeo, J. Broberg, and I. Brandic, "Cloud computing emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Comput. Syst., vol. 25, pp.599–616, 2009.

[3] Amazon Elastic Compute Cloud (Amazon EC2) [Online].vailable:http://aws.amazon.com/ec2

[4] Amazon Web Services (AWS) [Online]. Available:https://s3.amazonaws. com

[5] V. Goyal, O. Pandey, A. Sahai, and B.Waters, "Attibute based Encryption for fine-grained access control of encrypted data," in Proc. ACM Conf. Computer and Communications Security (ACM CCS), Alexandria,VA 2006.

[6] S. Yu, C. Wang, K. Ren, and W. Lou, "Achiving secure, scalable, and fine-grained data access control in cloud computing," in Proc. IEEE INFOCOM 2010, 2010, pp. 534–542.

[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext policy a Attribute based encryption," in Proc. IEEE Symp. Security and Privacy, Oakland,CA, 2007.

[8] R. Bobba, H. Khurana, and M. Prabhakaran, "Attribute sets: A practically motivated enhancement to attribute-based encryption," in Proc.ESORICS, Saint Malo, France, 2009