Attribute-Based Encryption Techniques in Cloud Computing Security : An Overview

Mr. Anup R. Nimje #1 , Prof. V. T. Gaikwad*2 ,Prof. H. N. Datir^3

[1]Research Student Of Master Of Engineering, Computer Engineering
[2]Asso. Professor and Head, Department of Information Technology
[3]Asst. Professor, Department of Computer Science and Engineering

Sipna's College Of Engineering and Technology, Amravati-INDIA

***Abstract*—Cloud computing is going to be very famous technology in IT enterprises. For an enterprise, the data stored is huge and it is very precious. All tasks are performed through networks. Hence, it becomes very important to have the secured use of data. In cloud computing, the most important concerns of security are data security and privacy. And also flexible and scalable, fine grained access control must be maintained in the cloud systems. For access control, being one of the classic research topics, many schemes have been proposed and implemented. There are policy based schemes have been proposed. In this paper, we are going to explore various schemes for encryption that consist of Attribute based encryption (ABE) and its types KP-ABE, CP-ABE. Further discussion consists of improvement in CP-ABE to CP-ASBE and to HASBE. A comparison table has been included for comparative study of these techniques.**

***Keywords*—Access control, Attribute based encryption, Key policy, ciphertext policy, hierarchical-ASBE.**

I. Introduction

Cloud computing technology consists of the use of computing resources that are delivered as a service over a network. In cloud computing model users have to give access to their data for storing and performing the desired business operations. Hence cloud service provider must provide the trust and security, as there is valuable and sensitive data in huge amount stored on the clouds. There are concerns about flexible, scalable and fine grained access control in the cloud computing.

For this purpose, there have been many of the schemes, proposed for encryption. Such as simple encryption technique that is classically studied. We are going to discuss about the Attribute-Based Encryption (ABE) schemes[1] and how it has been developed and modified further into Key Policy Attribute based encryption (KP-ABE) , Cipher-text Policy Attribute Based Encryption (CP-ABE) and further it has been proposed as CP-ASBE and furthermore HABE and HASBE so on. This is according to how flexible, scalable and fine grained access control [10] is provided by each scheme.

II. Literature Review

A. *Attribute based encryption (ABE):-*

Sahai and Waters [2] first introduced the attribute based encryption (ABE) for enforced access control through public key cryptography. The main goal for these models is to provide security and access control. The main aspects are to provide flexibility, scalability and fine grained access control. In classical model, this can be achieved only when user and server are in a trusted domain. But what if their domains are not trusted or not same? So, the new access control scheme that is 'Attribute Based Encryption (ABE)' scheme was introduced which consist of key policy attribute based encryption (KP-ABE). As compared with classical model, KP-ABE provided fine grained access control. However it fails with respect to flexibility and scalability when authorities at multiple levels are considered.

In ABE scheme both the user secret key and the cipher-text are associated with a set of attributes. A user is able to decrypt the cipher-text if and only if at least a threshold number of attributes overlap between the cipher-text and user secret key. Different from traditional public key cryptography such as Identity-Based Encryption [3], ABE is implemented for one-to many encryption in which cipher-texts are not necessarily encrypted to one particular user, it may be for more than one number of users. In Sahai and Waters ABE scheme, the threshold semantics are not very expressive to be used for designing more general access control system. Attribute-Based Encryption (ABE) in which policies are specified and enforced in the encryption algorithm itself. The existing ABE schemes are of two types. They are Key-Policy ABE (KP-ABE) scheme and Ciphertext-Policy ABE (CP-ABE) scheme. That can be discussed further.

B. *Key Policy Attribute Based Encryption(KP-ABE):-*

To enable more general access control, *V. Goyal, O. Pandey, A. Sahai, and B. Waters* [4] proposed a key-policy attribute-based encryption (KP-ABE) scheme. It is the modified form of classical model of ABE. Exploring KP-ABE scheme, attribute policies are associated with keys and data is associated with attributes. The keys only associated with the policy that is to be satisfied by the attributes that are associating the data can decrypt the data. Key Policy Attribute Based Encryption (KP-ABE) scheme is a public key encryption technique that is designed for one-to-many communications. In this scheme, data is associated with the attributes for which a public key is defined for each. Encrypter, that is who encrypts the data, is associated with the set of attributes to the data or message by encrypting it with a public key. Users are assigned with an access tree structure over the data attributes. The nodes of the access tree are the threshold gates. The leaf nodes are associated with attributes. The secret key of the user is defined to reflect the access tree structure. Hence, the user is able to decrypt the message that is a ciphertext if and only if the data attributes satisfy the access tree structure. In KP-ABE, a set of attributes is associated with ciphertext and the user's decryption key is associated with a monotonic *access tree structure* [5]. When the attributes associated with the ciphertext satisfy the access tree structure, then the user can decrypt the ciphertext.

In the cloud computing, for efficient revocation, an access control mechanism based on KP-ABE and a re-encryption technique used together. It enables a data owner to reduce most of the computational overhead to the servers. The KP-ABE scheme provides fine-grained access control. Each file or message is encrypted with a symmetric data encryption key (DEK), which is again encrypted by a public key, that is corresponding to a set of attributes in KP-ABE, which is generated corresponding to an access tree structure. The encrypted data file is stored with the corresponding attributes and the encrypted DEK. If and only if the corresponding attributes of a file or message stored in the cloud satisfy the access structure of a user's key, then the user is able to decrypt the encrypted DEK. That can be used to decrypt the file or message.

KP-ABE scheme consists of the following four algorithms:

1. ***Setup :*** This algorithm takes as input a security parameter κ and returns the public key PK and a system master secret key MK.
   PK is used by message senders for encryption. MK is used to generate user secret keys and is known only to the authority.
2. ***Encryption :*** This algorithm takes a message M, the public key PK, and a set of attributes as input. It outputs the ciphertext E.
3. ***Key Generation :*** This algorithm takes as input an access structure T and the master secret key MK. It outputs a secret key SK that enables the user to decrypt a message encrypted under a set of attributes if and only if matches T.
4. ***Decryption :*** It takes as input the user's secret key SK for access structure T and the ciphertext E, which was encrypted under the attribute set . This algorithm outputs the message M if and only if the attribute set satisfies the user's access structure T.

Limitations of KP-ABE:-

Encryptor cannot decide who can decrypt the encrypted data. It can only choose descriptive attributes for the data, and has no choice but to trust the key issuer. KP-ABE is not naturally suitable to certain applications. For example, sophisticated broadcast encryption [6], where users are described by various attributes and in this, the one whose attributes match a policy associated with a ciphertext, it can decrypt the ciphertext. KP-ABE scheme supports user secret key accountability. It is providing fine grained access but has no longer with flexibility and scalability.

C. *Expressive Key Policy Attribute Based Encryption:-*

In KP-ABE, enables senders to encrypt messages with a set of attributes and private keys are associated with access tree structure. Access tree structure specifies which all the ciphertexts the key holder is allowed to decrypt.

Expressive key-policy attribute-based encryption (KP-ABE) schemes allow for non-monotonic access structures. Non monotonic access tree structures are those may contain negated attributes and with constant cipher-text size. This is more efficient than KP-ABE.

D. *Cipher Text Policy Attribute Based Encryption:-*

Sahai et al.[8] introduced the concept of another modified form of ABE called CP-ABE that is Ciphertext Policy Attribute Based Encryption. In CP-ABE scheme, attribute policies are associated with data and attributes are associated with keys and only those keys that the associated attributes satisfy the policy associated with the data are able to decrypt the data. CP-ABE works in the reverse way of KP-ABE. In CP-ABE the ciphertext is associated with an access tree structure and each user secret key is embedded with a set of attributes. In ABE, including KP-ABE and CP-ABE, the authority runs the algorithm Setup and Key Generation to generate system MK, PK, and user secret keys. Only authorized users (i.e., users with intended access structures) are able to decrypt by calling the algorithm Decryption.

In CP-ABE, each user is associated with a set of attributes. His secret key is generated based on his attributes. While encrypting a message, the encryptor specifies the threshold access structure for his interested attributes. This message is then encrypted based on this access structure such that only those whose attributes satisfy the access structure can decrypt it. With CP ABE technique, encrypted data can be kept confidential and secure against collusion attacks.

CP-ABE scheme consists of following four algorithms:

1. ***Setup :*** This algorithm takes as input a security parameter κ and returns the public key PK as well as a system master secret key MK. PK is used by message senders for encryption. MK is used to generate user secret keys and is known only to the authority.
2. ***Encrypt :*** This algorithm takes as input the public parameter PK, a message M, and an access structure T. It outputs the ciphertext CT.
3. ***Key-Gen :*** This algorithm takes as input a set of attributes associated with the user and the master secret key MK. It outputs a secret key SK that enables the user to decrypt a message encrypted under an access tree structure T if and only if matches T.
4. ***Decrypt :*** This algorithm takes as input the ciphertext CT and a secret key SK for an attributes set . It returns the message M if and only if satisfies the access structure associated with the ciphertext CT.

In CP-ABE depends how attributes and policy are associated with cipher texts and users' decryption keys. In a

CP-ABE scheme, a ciphertext is associated with a monotonic tree access structure and a user's decryption key is associated with set of attributes.

In this scheme, the roles of ciphertexts and decryption keys are switched as that in KP-ABE. The ciphertext is encrypted with a access tree policy chosen by an encryptor. And the corresponding decryption key is created with respect to a set of attributes. As the set of attributes of a decryption key satisfy the access tree policy associated with a given ciphertext, the key can be used to decrypt the ciphertext. CP-ABE is conceptually closer to traditional access control models such as Role-Based Access Control (RBAC) [8] as users' decryption keys are associated with a set of attributes. Hence CP-ABE is more natural to apply instead of KP-ABE, to enforce access control of encrypted data.

Limitations of CP-ABE:-

However, basic CP-ABE schemes are still not fulfilling the enterprise requirements of access control which require considerable flexibility and efficiency. CP-ABE has limitations in specifying policies and managing user attributes. In a CP-ABE scheme, decryption keys only support user attributes that are organized logically as a single set, so users can only use all possible combinations of attributes in a single set issued in their keys to satisfy policies.

For realizing complex access control on encrypted data and maintaining confidential-ability, CP-ABE can be used. Encrypted data can be kept confidential even if the storage server is un-trusted; moreover, our methods are secure against collusion attacks. KP-ABE uses attributes to describe the encrypted data and built policies into user's keys. In other hand CP-ABE, attributes are used to describe a user's credentials. Data encryptor determines a policy for who can decrypt.

E. *Ciphertext Policy Attribute-Set Based Encryption (CP-ASBE):-*

As compared to CP-ABE scheme in which the decryption keys only support user attributes that are organized logically as a single set, so users can only use all possible combinations of attributes in a single set issued in their keys to satisfy policies.

To solve this problem, ciphertext-policy attribute-set-based encryption (CP-ASBE or ASBE for short) is introduced by *Bobba, Waters et al* [7]. ASBE is an extended form of CP-ABE which organizes user attributes into a recursive set structure. Ciphertext Policy Attribute Set Based Encryption (CP-ASBE) is a modified form of CP-ABE. It differs from existing CP-ABE schemes that represent user attributes as a monolithic set in keys. It organizes user attributes into a recursive set based structure and allows users to impose dynamic constraints on how those attributes may be combined to satisfy a policy. The CP-ASBE consists of recursive set of attributes. The desirable feature and the recursive key structure is implemented by four algorithms, *Setup, KeyGen, Encrypt, and Decrypt*[7] :

1. ***Setup:*** Here is the depth of key structure. Take as input a depth parameter 'd'. It outputs a public key PK and master secret key MK.
2. ***Key-gen:*** Takes as input the master secret key *MK*, the identity of user *u*, and a key structure *A* . It outputs a secret key SK for user u.
3. ***Encrypt:*** Takes as input the public key PK, a message M, and an access tree T . It outputs a ciphertext CT.
4. ***Decrypt:*** Take as input a ciphertext CT and a secret key SK for user u. It outputs a message m . If the key structure A associated with the secret key SK, satisfies the access tree T, associated with the ciphertext CT, then m is the original correct message M. Otherwise, m is null.

Specifically CP-ASBE allows-

User attributes are organized into a recursive family of sets and Allowing attributes to combine from multiple sets. Thus, by grouping user attributes into sets and no restriction on how they can be combined, CP-ASBE can support compound attributes. More flexibility and fine grained access is provided by AP-ASBE. Similarly, multiple numerical assignments for a given attribute can be supported by placing each assignment in a separate set as well as placing it into a single set.

Limitations:-

The challenge in constructing a CP-ASBE scheme is in selectively allowing users to combine attributes from multiple sets within a given key. There is challenge for preventing users from combining attributes from multiple keys.

F. *Identity Based Encryption (IBE) and Hierarchical Identity Based Encryption (HIBE) :-*

In an identity-based encryption scheme, data is encrypted using an arbitrary string as the key and for decryption; a decryption key is mapped to the arbitrary encryption key by a key authority. Hierarchical Identity Based Encryption (HIBE) is the hierarchical form of a single IBE [3]. The concept of HIBE scheme can help to explain the definition of security. In a regular IBE (1-HIBE) scheme; there is only one private key generator (PKG) that distributes private keys to each users, having public keys are their primitive ID (PID) arbitrary strings.

A two-level HIBE (2-HIBE) scheme consists of a root PKG, domain PKGs and users, all of which are associated with PID's. A user's public key consists of their PID and their domain's PID (in combine, called an address). In a 2-HIBE, users retrieve their private key from their domain PKG. Domain PKGs can compute the private key PK of any user in their domain, provided they have previously requested their domain secret key-SK from the root PKG. Similarly, is for number of sub-domains. There also includes a *trusted third party* or *root certificate authority* that allows a hierarchy of certificate authorities: Root certificate authority issues certificates for other authorities or users in their respective domains. The original system does not allow for such structure. However, a hierarchy of PKGs is reduces the workload on root server and allows key assignment at several levels.

For example, if the users of the system are employees of a group of companies, then each company is able to generate the private keys for their employees, so that employees request their keys from their company, rather than the top-level root PKG. Only companies can request only at once their domain secret from the top-level PKG.

G. Hierarchical Attribute-Base Encryption (HABE) and Hierarchical Attribute Set Based Encryption (HASBE):-

This scheme Hierarchical attribute-based encryption (HABE) is derived by Wang et al [9]. It is designed to achieve fine-grained access control in cloud storage services. It is a combination of HIBE and CP-ABE. In the HABE scheme, there are multiple keys with different usages. Therefore, we first provide a summary of the most relevant keys to serve as a quick reference. Then, HABE scheme is defined by presenting randomized polynomial time algorithms as follows:

1. ***Setup*** (K)→(params,MK0): The RM takes a sufficiently large security parameter K as input, and outputs system parameters params and root master key MK0.
2. ***CreateDM***(params,MKi, PKi+1) → (MKi+1): Whether the RM or the DM generates master keys for the DMs directly under it using params and its master key.
3. ***CreateUser***(params,MKi, PKu, PKa) → (SKi,u, SKi,u,a): The DM first checks whether U is eligible for a, which is administered by itself. If so, it generates a user identity secret key and a user attribute secret key for U, using params and its master key; otherwise, it outputs "NULL".
4. ***Encrypt***(params; f ;A; {PKa|a E A})→(CT): A user takes a file f, a DNF access control policy A, and public keys of all attributes in A, as inputs, and outputs a ciphertext CT.
5. ***Decrypt***(params, CT, SKi,u, {SKi,u,a|aECCj}→(f):A user,whose attributes satisfy the j-th conjunctive clause CCj, takes params, the ciphertext, the user identity secret key, and the user attribute secret keys on all attributes in CCj, as inputs, to recover the plaintext.

However, HABE uses disjunctive normal form policy. It assumes all attributes in one conjunctive clause those are administrated by the same domain master. Thus the same attribute may be administrated by multiple domain masters according to specific policies, which is most complicated to implement in practice. This scheme has issues with multiple values assignments. HASBE scheme is proposed and implemented by *Zhiguo Wan et al [10]*. The cloud computing system consists of five types of parties: a cloud service provider, data owners, data consumers, a number of domain authorities, and a trusted authority. The cloud service provider manages a cloud and provides data storage service. Data owners encrypt their data files and store them in the cloud for sharing with data consumers.

To access the shared data files, data consumers download encrypted data files of their interest from the cloud and then decrypt them. Each data owner/consumer is administrated by a domain authority. A domain authority is managed by its parent domain authority or the trusted authority. The HASBE scheme extends the ASBE scheme to handle the hierarchical structure of system as shown in figure-1.

The trusted authority is responsible for managing top-level domain authorities. It is root level authority. For example, for an IT enterprise, employees are kept in the lowest domain level and above that there is department and above that there is top level of domain we call it as a trusted domain. It generates and distributes system parameters and also root-master keys. And it authorizes the top-level domain authorities. A domain authority delegates the keys to its next level sub-domain authorities. Each user in the system is assigned a key structure. Key specifies the attributes associated with the user's decryption key. *Zhiguo Wan et al* [10] given a HASBE scheme for scalable, flexible, and fine-grained access control in cloud computing. The HASBE scheme consists of hierarchical structure of system users by using a delegation algorithm to CP-ASBE. HASBE supports compound attributes due to *flexible attribute set combinations* as well as achieves efficient user revocation because of attributes assigned multiple values. Thus, it provides more scalable, flexible and fine grained access control for cloud computing.

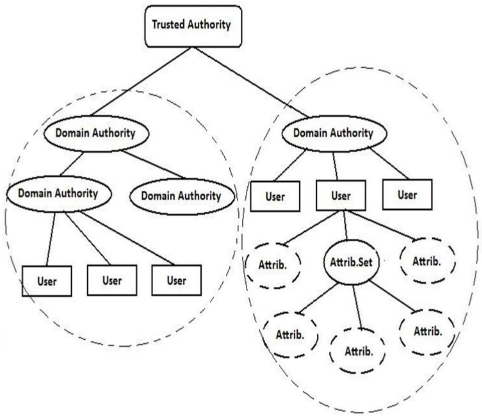

FIGURE 1USERS HIERARCHY, DOMAINS, USERS AND ATTRIBUTES

III. CONCLUSION

In this paper we have overviewed different attributes based encryption (ABE) schemes that can be used in cloud systems for flexible, scalable and fine grained access control. In ABE scheme, there are both the 'secret key' and 'cipher-text' are associated with a set of attributes. ABE is further modified into KP-ABE that provides fine grained access control. In KP-ABE, attribute policies are associated with keys and data is associated with the attributes. Keys associated with the policy that is satisfied by the attributes can decrypt the data. Moreover, we have explored CP-ABE and CP-ASBE. The CP-ABE scheme differs from KP-ABE in such a way that in CP-ABE, ciphertext is associated with an 'access tree structure' and each user 'secret key' is embedded with a 'set of attributes'. Attribute policies are associated with data and attributes are associated with keys and only those keys that the associated attributes satisfy the policy associated with the data are able to decrypt the data.

HASBE combines the functionalities of HIBE and ASBE. HASBE scheme seamlessly incorporates a hierarchical structure of system users. It uses a delegation algorithm to

ASBE. Out of these schemes, the HASBE scheme provides more scalable, flexible and fine-grained access control than any other schemes in cloud computing.

REFERENCES

[1] D.Boneh and M. Franklin. "Identity-Based Encryption from the Weil Pairing." *In Proc. of CRYPTO'01, Santa Barbara, California, USA, 2001.*

[2] J.Bethencourt, A. Sahai, and B. Waters. "Ciphertext-Policy Attribute-Based Encryption." In Proc. of SP'07, Washington, DC, USA, 2007.

[3] A.Sahai and B. Waters. "Fuzzy Identity-Based Encryption." In Proc. of EUROCRYPT'05, Aarhus,Denmark, 2005.

[4] V. Goyal, O. Pandey, A. Sahai, and B. Waters. "Attribute-Based Encryption for Fine-grained Access Control of Encrypted Data". In Proc. of CCS'06, Alexandria, Virginia, USA, 2006.

[5] R.Ostrovsky, A. Sahai, and B. Waters. "Attribute-based encryption with non-monotonic access structures". In Proc. of CCS'06, New York, NY, 2007.

[6] Zhibin Zhou, Dijiang Huang" On Efficient Ciphertext-Policy Attribute Based Encryption and Broadcast Encryption"

[7] Rakesh Bobba, Himanshu Khurana and Manoj Prabhakaran, "Attribute-Sets: A Practically Motivated Enhancement to Attribute-Based Encryption", July 27, 2009

[8] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-policy attribute based encryption," *IEEE Symp. Security and Privacy*, Oakland, CA, 2007.

[9] G.Wang, Q. Liu, and J.Wu, "Hierachical attibute-based encryption for fine-grained access control in cloud storage services," in *Proc. ACM Conf. Computer and Communications Security (ACM CCS)*, Chicago, IL, 2010.

[10] Zhiguo Wan, Jun'e Liu, and Robert H. Deng, "HASBE: A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing" *IEEE Transactions On Information Forensics And Security*, Vol. 7, No. 2, April 2012

APPENDIX

TABLE 1: COMPARISON REVIEW OF VARIOUS ABE TECHNIQUES

Parameters v/s ABE-Technique	Fine-grained access control	Efficiency	Computational Overhead
KP-ABE	Low, High if there is re-encryption technique	Average High for broadcast type system	Most of computational overheads
EKP-ABE	Better Access control than that of KP-ABE	Higher than KP-ABE, allows constant cipher text only	Reduces computational overheads
CP-ABE	Average Realization of complex Access Control	Average Not efficient for modern enterprise environments	Average computational overheads
CP-ASBE	Better Access Control than that of CP-ABE	Better than CP-ABE as there is Less collusion attacks	Lower than CP-ABE computational overheads
HIBE	Lower than CP-ASBE	Better, Lower as compared to ABE schemes	Most computational overheads
HABE	Good Access control	Flexible and scalable	Some of overhead
HASBE	Better Access control	Most efficient and flexible	Less overhead than others