A Survey on Data Outsourcing in Cloud by Security Enhanced CP-ABE Access Control Scheme

Mr. Rahul Mahavir Patil[1], Mr. Dhainje Prakash.B.[2]
Department of CSE
[1]Student- ME,Shriram Institute of Engineering & Technology Center, Paniv, Maharashtra, India
[2]Vice-Principal,Shriram Institute of Engineering & Technology Center, Paniv, Maharashtra, India

Abstract-*In several distributed systems a user should only be able to access data if a user posses a certain set of credentials or attributes. Security is an important property in role based access control system. To preserve the confidentiality and Integrity we need certain policies. We present a system for realizing complex access control on encrypted data that we call Cipher text-Policy Attribute-Based Encryption. We enhance abstract syntax tree (express the user role and permission) with cipher text policy attribute based encryption technique.*

This will provide the security policy for the administration. The cipher text policy attribute based encryption is much more flexible than plain identity-based encryption. The access control to system resources must be provided efficiently and confidentially. The security policy such as Organization policy and Government policy can be made. Proposes an access control scheme called Ciphertext-Policy Attribute Role Based Encryption (CP-ARBE). Our CP-ARBE integrates Role based Access Control (RBAC) into a Ciphertext-Policy Attribute-based Encryption (CP-ABE).

Keywords-access control; privacy; collaborative cloud; key management; user revocation; attribute-based encryption

I. INTRODUCTION

The CP-ABE is much more flexible than plain identity-based encryption, in that it allows complex rules which specify private keys can decrypt cipher text. The identity-based encryption can be done by using one public key and Master Private Key used to make more restricted private keys. But there is very expressive rules for which private keys can decrypt .They are Private keys have “attributes” or labels and Cipher texts have decryption policies. CP-ABE can be viewed as a generalization of identity-based encryption. So as in identity-based encryption, there is a single public key, and there is a master private key that can be used to make more limited private keys. Using a few keys to encrypt many files may lose the fine grained control we had over access policies.

However, specifically, the private keys are associated with sets of attributes or labels, and when we encrypt, we encrypt to an access policy which specifies which keys will be able to decrypt. To access any files in remote storage the scalable and reliability is an important property. The more we replicate our files, the more we introduce potential points of compromise and the more trust we require. For this problem CP-ABE may be useful. The abstract syntax has been extended to include set and set operations to model role-based access control requirements. A set can be used as a component itself or can be an attribute of a component. Basic set operations such as union, compliment, and intersection are allowed. It also includes membership and cardinality. A session can be represented as a component and may have active roles as an attributes. Objects are represented as components and operations performed on them are represented as its states

We integrate RBAC model into CP-ABE to provide a more expressiveness of policy specification. In the policy tree structure, the operations AND, OR, and K of N are supported to logically express the natural evaluation for roles and attributes as the access control rules. The policy also accommodates the privilege (read or write) of users for each role distinctively. User attributes from multiple domains can be specified under the respective policy of any data owners. We demonstrate the efficiency and practicality of the access control features through the functionality evaluation. Regarding the scalability, the proposed model enables CP-ABE policy tree to support a more large number of users and better attribute management by assigning a group of attributes belong to the specific role. This enables the model is scalable in terms of multiple user management. In addition, we exploit user decryption key graph (UDKG) to make all user decryption keys are securely stored in a cloud. User keys will be dynamically invoked upon the user’s request for access. This provides zero cost for key distribution and enables efficient multiple keys assignment and retrieval. This is a desired feature that could make our proposed model is practical in a large scale of data sharing environment where there is multi-user, multi-owner, and multi-authority. In contrast, approaches based on CP-ABE require distributing every user decryption keys to all individual users who request for the key. The cost for key delivery therefore depends on

network conditions and is linear to the number of registered users.

II. LITERATURE REVIEW

Attribute-based encryption (ABE) is regarded as a suitable solution for formulating a light-weight access control to outsourced data. The key construction of attribute-based encryption is based on bilinear maps. An attribute based encryption scheme introduced by Sahaiand Waters in 2005 and the goal is to provide security and access control. Attribute-based encryption (ABE) is a public-key based one to many encryptions that allows users to encrypt and decrypt data based on user attributes. In which the secretkey of a user and the ciphertext are dependent upon attributes (e.g. the country she lives, or the kind of subscription she has). In such a system, the decryption of a ciphertext is possible only if the set of attributes of the user key matches the attributes of the ciphertext. Decryption is only possible when the number of matching is at least a threshold value d. Collusion-resistance is crucial security feature of Attribute-Based Encryption .An adversary that holds multiple keys should only be able to access data if at least one individual key grants access.

Goyal et al. proposed key-policy attribute-based encryption (KP-ABE) to serve a more general and richer encrypted access control. In this scheme, the ciphertext is associated with a set of attributes for each of which a public key component is defined. User secret key is constructed to associate with the access structure. However, the KP-ABE-based schemes do not give the data owner has a full control over the access policy to reflect the access tree Structure the secret key of the user is defined. Ciphertexts are labeled with sets of attributes and private keys are associated with monotonic access structures that control which ciphertexts a user is able to decrypt. Key Policy Attribute Based Encryption (KP-ABE) scheme is designed for one-to-many communications

To address this drawback, the Ciphertext Policy Attribute Based Encryption (CP-ABE) was proposed. In CPABE, the ciphertext is associated with the access policy structure in which the encrypt or can define the access policy by her own control. Users are able to decrypt a ciphertext if their attributes satisfy the ciphertext access structure.

A role-based encryption (RBE) scheme for cloud storage systems. The proposed RBE uses ABE for cryptographic access control and use identity broadcast encryption for key distribution. For the access control, the role-based access control (RBAC) policy is enforced through a public parameter of role and a group public to encrypt the data. However, in this system data is encrypted by the data owner to the specific role, several copies of the encrypted data are required for users in different roles. To support multi-owner and multi-authority cloud, a multi authority attribute based encryption (MA-ABE) is recently proposed by several works.

Hierarchical attribute-set-based encryption (HASBE) by extending ciphertext-policy attribute set- based encryption (ASBE) with a hierarchical structure of users. In this scheme, a trusted authority is responsible for generating and distributing system parameters and root master keys as well as authorizing the top-level domain authorities. Each user in the system is assigned a key structure which specifies the attributes associated with the user's decryption key. However, the vulnerability of trusted authority of the hierarchical domains would be at risk to all users.

Kan Yang et al proposes DAC-MACS (Data Access Control for Multi-Authority Cloud Storage model. The authors apply CP-ABE technique to construct an access control model where there are several multi-authority issuing the attributes. The proposed scheme improves the decryption process and solves revocation problem in ABE by designing the decryption token and key update and ciphertext update algorithms. For the immediate revocation, their scheme reduces the cost for data re-encryption since only the ciphertext getting an effect is updated.

III. METHODOLOGY

Our proposed cryptographic process of C-CP-ARBE scheme is based on the bilinear map used in traditional ABE. Our system model which consists of the following four entities and our access control mechanism. Collaborative Access Control in Multi-Authority Cloud Storage Systems

1. Authentication Module (AM)
   This module is the first gate to control the access of any entities to the data resided in the cloud storage.

2. User-Role Management Module (URMM)
   This module, all authorized users are registered and mapped to the role. Under the role, attributes associated to users are grouped to describe the role of the users. This module maintains the role and attributes constructs. Public role parameter of each role is generated to represent as a public value shared among the users of the role.

3. Crypto Module (CM)

The Crypto module is a core engine of our proposed scheme. At a nutshell of this model, we propose two encryption layer called SEAL (Secret Encryption over Attribute-based Encryption Layer) to support strong encryption and enable the optimization for key management and user revocation cost.

4. Multi-AA and Access Control Policy Management Module (MACPM)

   This module controls multiple attributes issued by multiple attribute authorities (AAs). Each AA also has its own public key, private key and certificates obtained from trusted CAs.

IV. SYSTEM SETUP

1. Create Attribute Authority
   - The algorithm takes the attribute authority IDas input.
   - It outputs the authority public key (public parameter) and public attribute keys for all attributes issued by the Attribute Authority.
   - Attribute Authority has also a key pairs (publickey, IDprivate keyID) generated by CA.

2. UserRegister.
   - The userRegister algorithm takes input as userID and user's certificate issued by a trusted CA.
   - If the user is authorized, the user list associated to particular secret key role is updated.

3. CreatRole.
   - The CreateRole algorithm takes as inputs attribute authority's secret keyIDSKaid ,
   - RoleID RID, and set of users UID who belong to the role.
   - It returns master key of role MKR, and user list UL.

4. Create GroupRole parameter.
   - The Create GroupRole parameter algorithm takes input as a set of RID and returns the GRP.
   - Then, the GRP is signed (encrypted) by AA's private key and it will be updated when there is any adding or revoking of user to/from any role.

5. Create UDKG.
   - The algorithm takes set of user who uses the resource in the authority domain. It outputs the user decryption key graph (UDKG) with the root node labeled as the user id and empty key node.
   - UDKG is a graph structure used to store the UDKs encrypted by each user's private key. Hence, all decryption keys are not distributed to users but they are stored in a cloud.

V. KEY GENERATION

1. UserKeyGen

   The KeyGen algorithm takes continuous two steps
   - Takes input as set of attributes attribute authority's secret key , and public key certificate of users , then it returns the set of user decryption keys UDK
   - A UDK is encrypted with the global public key of the user and outputs the set of encrypted decryption keys.

2. UpdateUDKG
   - This algorithm takes user id and encrypted decryption key (EDK) to update the UDKG.

Encryption

This phase runs our two encryption layer protocol which accommodates the following two algorithms

1. Enc

The encryption algorithm performs two continuous steps as followings:

- **Inner layer**: the algorithm takes as inputs authority public key PKaid, access control policy ACP, and data M. Then it returns a ciphertext CT.
- **Outer Layer**: the algorithm takes GRP and generates 3DES session key as a secret seal SS to encrypt the ciphertext CT. It returns sealed ciphertextSCT.

2. EncSeal.
   - The algorithm takes inputs as secret seal SS and then encrypts the SSwith the Certuid and publishes the encrypted secret seal to the user decryption key graph UDKG andstored in pair with the EDKvid,uid,aid.

Decrypt

The decryption algorithm performs two continuous steps as follows:
- Decrypt the secret seal SS. The algorithm takes user's global secret key GSKuid and then obtains the session key to decrypt the SCT and get the CT.
- Decrypt the encrypted decryption key(EDKuid). The algorithm takes user's global secret keyGSKuid and then obtains the user decryption key UDK. Together the PKaid, if the set of attribute S satisfies the ACP structure, the algorithm returns the message M.

VI. CONCLUSION

We could conclude that our paper describes the security policy for an organization. In this paper we built abstract syntax tree for a role access and to secure the process CP ABE algorithm is used. For any role which has been accessed by user is secured and prevented by CP ABE algorithm. The encryption and decryption technique is used in which key has been protected. In our future work we would like to extend our security policy.

We have presented our proposed access control scheme CCP- ARBE which is based on the combination of CP-ABE and Role-based access control model. The proposed scheme achieves the scalable and efficient collaborative data sharing in multi-authority cloud data storage systems. Specifically, our two-layer encryption strategy provides a significant improvement for key distribution and user revocation.

REFERENCES

[1] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Encryption Policies for Regulating Access to Outsourced Data", in ACM Transactions on Database Systems (TODS), April, 2010.
[2] Shucheng Yu, Cong Wang, KuiRen, and Wenjing Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing," IEEE INFOCOM 2010, San Diego, CA, March, 2010.
[3] Zhiguo Wan, Jun-e Liu, 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 7(2): 743-754, 2012.
[4] Ming Li, Shucheng Yu, Yao Zheng, KuiRen, and Wenjing Lou, "Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption," IEEE Transactions on Parallel and Distributed Systems, 2012
[5] Kan Yang, XiaohuaJia, KuiRen, Bo Zhang, RuitaoXie: DAC-MACS: Effective Data Access Control for Multiauthority Cloud Storage Systems., IEEE Transactions on Information Forensics and Security 8(11): 1790-1801, 2013.
[6] 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.
[7] M. Chase, "Multi-authority attribute based encryption", in Proceedings of the 4th Theory of Cryptography Conference on Theory of Cryptography (TCC'07), Springer, 2007.
[8] M. Chase and S. S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS'09), ACM, 2009.