IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 04, Issue 09 (September. 2014), ||V3|| PP 21-25

Session Based Ciphertext Policy Attribute Based Encryption Method for Access Control in Cloud Storage

Priyanka Rajput, Pankaj Kwadakar
Dept. of Computer Science & Engineering, Patel College of Science & Technology, Bhopal, India
Professor, Dept. of Computer Science & Engineering, Patel College of Science & Technology, Bhopal, India

Abstract: - Everyday operations of modern enterprises heavily depend on their information processing capability, and the costs and overheads of managing them bring about the serious challenges. In order to relieve the burdens of IT resources and of maintaining them by professionals, cloud computing is introduced. CLOUD computing is a new computing paradigm that is constructed on virtualization, aligned and circulated computing, utility computing, and service oriented architecture. The advantages of cloud computing include decreased charges and capital expenditures, expanded operational efficiencies, scalability, flexibility, immediate time to market, and so on. Cloud storage is a basic service which has been widely used today. However, due to network security or trust to Cloud Storage service Providers (CSP), and other security incidents, many individuals and businesses are afraid to upload their own important or private data to the cloud storage servers. Thus, the access control security of cloud storage has become the focus of research in cloud storage filed. In this paper we propose a deployment model (SB-CP-ABE) for ABE which enables management of access rights as well as efficient key refreshing and revocation.

Keywords: - *Cloud Storage; Access Control; Attribute-Based Encryption;Document Sharing;Ciphertext-Policy.*

I. INTRODUCTION

Cloud storage is a service based on cloud computing technology [1]. Storage virtualization consolidates different storage resources that can be accessed through a single user interface via the Internet without exposing the physical details of the underlying infrastructure. Cloud storage has the capability of providing almost unlimited flexible storage capacity and rapid provisioning to users, as well as dramatically reducing the costs of IT ownership and maintenance.

Data outsourcing to third party cloud storage providers presents a number of issues. In order to provide virtually unlimited storage resources to end users, a cloud storage service usually spans multiple domains. Thus, data from different logical domains may be hosted at the same physical or virtual server, or the data may be segmented and stored on multiple servers across different security domains. Since virtualization hides the details of physical resources, the location of stored data becomes uncertain to users, which has a potential to result in mistrust of cloud storage service providers [2], [3].

From a data security perspective, data owners should take responsibility for protecting their own data. This data owner-centric protection approach typically requires the following characteristics:

a. Fine-grained protection. Data access policy can be defined at data item level. The data access policy should be enforced at each access attempt with or without the data owner's involvement.
b. Dynamic access rights management. The granting or revoking of access rights to a particular item of data is straightforward to conduct and can, ideally, be performed almost instantaneously.
c. Efficient key management. Critical key management operation such as key establishment, key refreshing and key revocation are conducted in an efficient manner that scales well and is appropriate for the highly dynamic and heterogeneous nature of a cloud storage environment.

Recently, attribute-based encryption (ABE) has been developed as a cryptographic primitive for the provision of fine-grained access control to encrypted data. In ABE, a set of system attributes are used to define user access rights or data access policies. ABE thus appears to be a promising tool for the protection of data in cloud storage environments. However, existing ABE schemes have some practical limitations with respect to the efficiency and scalability of certain operations that are critical to cloud storage environments, in particular revocation of access rights, key refreshing and revocation. In this paper we propose a deployment model (SB-CP-ABE) for ABE which enables management of access rights as well as efficient key refreshing and revocation. This model can be generically adapted to suit ciphertext-policy ABE (CP-ABE) schemes.

The rest of paper is organized as follows. Section II provides a brief background of Cloud Storage Trust Model. In Section III we discussed about ABE and discussion of related work is happened in section IV. We explain the Proposed Method- Session Based Cipher Policy-Attribute Based Encryption- in Section V and apply it to an existing CP-ABE and proposed SB-CP-ABE scheme in Section VI. We draw our conclusions in Section VII.

II. CLOUD STORAGE TRUST MODEL

In our model, the attribute authority is the central trusted component that is responsible for generating attribute key shares, publishing system public parameters and maintaining the master secret. The cloud storage provider, which includes a proxy server, is a semi-trusted entity. It is responsible for re-encrypting data owners' cipher texts before they are sent to users. Data owners are responsible for protecting their data by defining access policies, managing user revocation lists, and encrypting data before it is sent to the cloud storage provider. Users are untrusted entities whose attributes need to comply with the access policy before the data is able to be decrypted. All the communication channels need to be encrypted for data transmission.

III. ATTRIBUTE-BASED ENCRYPTION

ABE was first introduced by Sahai and Waters [4]. There are two major classes of ABE schemes. In key policy ABE (KP-ABE) [5], [6] cipher texts are labeled with sets of attributes and private keys are associated with access policies. A user can decrypt a ciphertext if the attributes associated with the ciphertext satisfy the access policy associated with the private decryption key. In ciphertext-policy ABE (CP-ABE) [7], [8], [9] an access policy is associated with each ciphertext. The private decryption key can be reconstructed correctly if a user's attributes satisfy the access policy. Although KP-ABE and CP-ABE both achieve fine-grained access control, CPABE is more suitable for data-owner-centric protection in outsourcing systems.

IV. RELATED WORK

There has been some prior research into dealing with the practical problems with implementation of ABE, particularly with respect to revocation issues.

In [10] it was suggested that attributes could be associated with an expiry time. This idea was enhanced by [7] who suggested associating private key shares with an expiry time. Both the resulting schemes require users to periodically contact the attribute authority for generation of new key shares. This raises potential issues of scalability with both schemes, as well as the problem that user revocation cannot be instantaneous.

In [11] Junod and Karlov constructed a CP-ABE based broadcast encryption scheme that supports direct user revocation. In their scheme, each broadcast receiver's identity is mapped to an individual attribute. The access policy consists of a set of system attributes with a set of identity attributes. Individual user revocation is achieved by updating the set of identity attributes in the access policy. This scheme is not efficient to apply to cloud storage systems as mapping each user's identity to an attribute can make the ciphertext growing linearly. In addition, data owners should not directly control the data distribution after the data is stored in a cloud storage system.

Yu et al. [12] proposed a scheme to accomplish revocation of user access rights via attribute revocation. One of their core mechanisms is proxy re-encryption, which was first introduced in [13]. The notion of the proxy re-encryption is to use a proxy to re-encrypt a ciphertext from one secret key to another without learning the underlying plaintext. In the scheme of [12], when a user's access right is revoked, the attribute authority generates a new re-encryption key for the system's semi-trusted on-line proxy server. On behalf of the attribute authority, the proxy server generates and distributes new updated attribute key shares for each non-revoked user. Then the proxy server re-encrypts the existing cipher-texts with the new re-encryption key. While this scheme enables instantaneous user revocation, each revocation triggers a round of attribute key share updates and ciphertext re-encryption. This results in it being unsuitable for large data-owner-centric environments.

Jahid et al. in [14] achieved user revocation by utilizing a semi-trusted proxy to participate in the decryption process. In their proposed scheme, each user obtains an identity key in addition to their attribute key shares. The identity keys are generated by a data owner using a secret sharing scheme. The data owner also generates a proxy key for the proxy, who uses the proxy key to transfer the ciphertext in the way such that only non-revoked users with their identity keys can decrypt the data. The proxy key is regenerated whenever a user is revoked. Although the scheme achieves dynamic user revocation without attribute key regeneration, it can only revoke up to a predefined number of users. In addition, adding a new user to the system can trigger the rekeying of existing users' identity keys, which presents a potential scalability issue in a large or highly distributed environment.

Hur and Noh [15] use attribute key encryption keys (KEKs) to address user revocation in BSW's CP-ABE scheme [7]. Their scheme requires a data service manager (such as the storage service provider) to generate attribute KEKs and distribute the keys to users. The attributes in the access policy of a ciphertext are re-encrypted by their KEKs before the ciphertext is sent to a requestor.

When a user is revoked, the impacted attribute KEKs are updated and redistributed. This approach brings potential management overheads and scalability issues. The attribute KEKs are generated and maintained via a global binary tree that assigns users to the leaf nodes. For a large group of users, maintaining the binary tree becomes much harder when the system needs to add or delete users. The data service manager also has to know every user's attribute set in order to generate and distribute their attribute KEKs, which can leak too much

information to a semi-trusted data service manager. In addition every user needs to have two sets of keys: secret attribute key shares and attribute KEKs.
Shuaishuai Zhu, Xiaoyuan Yang and XuGuang Wu [16] work on secure and practical attribute based encryption scheme without pairings (CP-ABE-WP) under cloud computing scenarios. It presents a new practical Attribute based Encryption Scheme without Pairing (CP-ABE-WP). Based on this, a secure file sharing system (SFSS) with attribute computing support is design.

V. PROPOSED SCHEME –-SESSION BASED CP-ABE

Access structure: Let G be a multiplicative cyclic group of prime order p. Let g be a generator of G and e be a bilinear map. Let H: {0, 1}*→ G be a hash function. Let K be the threshold of the access tree to control the amount of the shared group. Z_p be the Lagrange coefficient.
Setup: The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters PK and a master key MK.
Randomly choose two numbers, a1, a2 ε Z_p, and compute
PK = (G, g, h=g^{a2}, t=g^{a1})
MK = (a1, a2)
Generate session key (K_s).

Encryption (PK, A, M):

A encryption algorithm run by a sender. The encryption algorithm takes as input the public parameters PK, a message M, and an access structure A over the universe of attributes. The algorithm will encrypt M and produce a cipher-text CT such that only a user that possesses a set of attributes that satisfies the access structure will be able to decrypt the message. Assume that the cipher-text implicitly contains A. Outputs the cipher-text. Mathematically express as:
Firstly, starting from the root node, choose a polynomial q_x with order of d_x for each node x on the tree, and let d_x= k_x−1 to generate node specific key (Ki).
Secondly, From the root node, randomly choose a number s ∈ Z_p and let q_R (0)= s. The value of q_R on the other d_R is randomly picked.
Thirdly, from the size of file generate unique IDi.
Finally, let Y be the leaf node set of the access tree, and E_T be the ciphertext embedded into the access tree T. Then E_T can be computed by:
E_T= (T, C'=m.t^s, C=h^s, IDi, Ki)

Key Generation (MK, S):
The key generation algorithm takes as input the master key MK and a set of attributes s that describe the key and the public parameters PK1. It outputs a private key SK.
The secret key can be computed by:
SK= (D'=$g^{(a2+PK1).s}$)

Decryption (PK, CT, SK):

The decryption algorithm takes as input the public parameters PK, a ciphertext CT, which contains an access policy A, Generate session key(K_s), specific key (Ki), , choose unique IDi and a private key SK, which is a private key for a set s of attributes. If the set s of attributes satisfies the access structure A then the algorithm will decrypt the ciphertext and return a message M.

VI. RESULT AND ANALYSIS

To evaluate the feasibility of our proposed design, we conducted several experiments to measure overhead in terms of time. The experiments were run on a virtual machine with 1GB of RAM, and hosted on PIV. It also uses the Pairing Based Cryptography to perform the pairing-based cryptosystems utilizing bilinear mapping.
Time Overhead:
To measure the efficiency of the encryption and decryption algorithms, five texts (200 B, 400MB, 600B, 800B and 1000B) were encrypted with four different numbers of attributes in the access structure, and then decrypted with a secret key that is associated with same attributes.
The result of our method Session Based Cipher Policy –Attribute Based Encryption and CP-ABE-WP [16] conducted experiment is as follows:

Table 1: Comparison between CPU execution time of both algorithms.

S. No.	Size of Input (In Characters)	CP-ABE - WP (CPU Execution Time) in Nano Seconds	Session Based CP-ABE(CPU Execution Time) in Nano seconds	Diff-erence between Time
1	100	287332	128093	159239
2	200	289945	135939	154006
3	300	290677	145004	145673
4	400	308800	164607	144193
5	500	313242	163370	149872

This result can be interpreted like fig 1. IT shows the performance of both methods. It clearly shows that the performance of the Session-Based-CP-ABE is much better and takes lesser execution time to encrypt the data through the method.

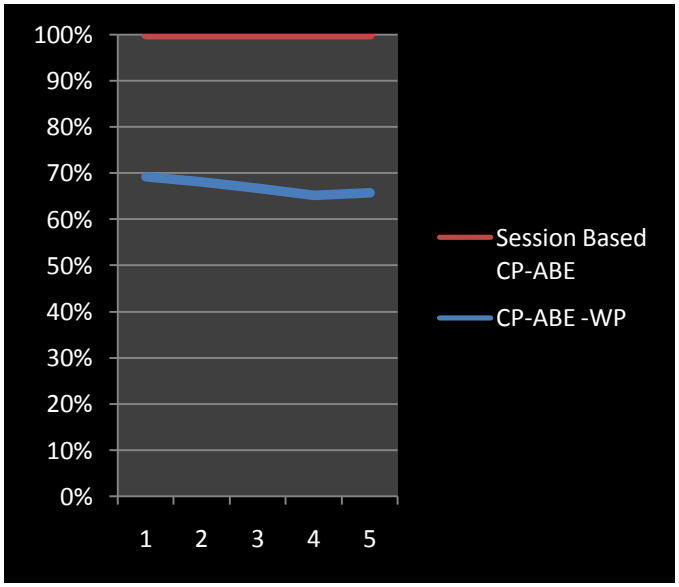

Fig 1: CPU Execution time of CP-ABE-WP and Session Based-CP-ABE methods.

VII. CONCLUSIONS

This paper presented a design for a secure cloud-based system. Although CP-ABE schemes provide the ability for data owner-centric protection in cloud storage, they are not very practical with the respect to the efficiency and scalability of access right revocation, key refreshing and revocation. Our proposal is efficient, scalable and has no limitation on the number of revoked users. It does require user attribute key shares to be re-generated or ciphertext to be re-encrypted. It also investigated the feasibility of adopting CP-ABE in terms of performance. The results suggest that the proposed design would provide reasonable performance, and thus it can be used as a replacement to standard encryption mechanisms in cloud based exiting systems.

REFERENCES

[1] W. Zeng, Y. Zhao, K. Ou, and W. Song, "Research on cloud storage architecture and key technologies," in2nd International Conference on Interaction Sciences: Information Technology, Culture and Human, 2009, pp. 1044–1048

[2] X. Jing and Z. Jian-jun, "A brief survey on the security model of cloud computing," in Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science, 2010, pp. 475 – 478.

[3] R. Chandramouli and P. Mell, "State of security readiness," in Crossroads. ACM, 2010, pp. 23 – 25.

[4] A. Sahai and B. Waters, "Fuzzy identity based encryption," in Advances in Cryptology, vol. 3494 of LNCS. Springer, 2005, pp. 457 – 473.
[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in 13th ACM conference on Computer and communications security. ACM, 2006, pp. 89 – 98.
[6] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in 14th ACM conference on computer and communications security. ACM, 2007, pp. 195 – 203.
[7] J. Bethen court, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in 2007 IEEE Symposium on Security and Privacy. IEEE Computer Society, 2007, pp. 321–334.
[8] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in ICALP, 2008, pp. 579 – 591.
[9] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in IACR Cryptology ePrint Archive, no. 290, 2008.
[10] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, "Secure attribute-based systems," in 13th ACM conference on computer and communications security, 2006, pp. 99 – 112.
[11] P. Junod and A. Karlov, "An efficient public-key attribute-based broadcast encryption scheme allowing arbitrary access policy in Tenth annual ACM workshop on digital rights management. ACM, 2010, pp. 13–24.
[12] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in ACM Symposium on Information, Computer and Communications Security, 2010, pp. 261 – 270.
[13] M. Blaze, G. Bleumer, and M. Strauss, "Divertable protocols and atomic proxy cryptography," in EUROCRYPT, vol. 1403 of LNCS. Springer, 1998, pp. 127–144.
[14] S. Jahid, P. Mittal, and N. Borisov, "Easier: Encryption-based access control in social networks with efficient revocation," in 6th ACM Symposium on Information Computer and Communications Security, 2011, pp. 411–415.
[15] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," in IEEE Transactions on Parallel and Distributed Systems, vol. 22, 2011, pp. 1214 – 1221.
[16] Shuaishuai Zhu, Xiaoyuan Yang and XuGuang Wu, "Secure Cloud File System with Attribute based Encryption," 2013 5th International Conference on Intelligent Networking and Collaborative System 2013 IEEE.