ISSN : 2394-2975 (Online)
ISSN : 2394-6814 (Print)
Vol. 2, Issue 3 (July - Sept. 2015)
International Journal of Advanced Research in Education & Technology (IJARET)

Fully Secure and Efficient Data Sharing with Attribute Revocation for Multi-Owner Cloud Storage

[I]K.Kokilavani, [II]K.S. Saravanan
[I]M.Phil Full Time Research Scholar, Dept. of Computer Science
[II]Assistant Professor, Dept. of Computer Sc. and Application
[I,II]Vivekananda College of Arts and Sciences for Women, Namakkal, TamilNadu, India

Abstract

Now a days, a lot of users are storing their data's in cloud, because it provides storage flexibility. But the main problem in cloud is data security. Data access control is an effective way to ensure the data security in the cloud. Due to data outsourcing and untrusted cloud servers, the data access control becomes a challenging issue in cloud storage systems. Ciphertext-Policy Attribute based Encryption (CP-ABE) is regarded as one of the most suitable technologies for data access control in cloud storage, because it gives data owners more direct control on access policies. However, it is difficult to directly apply existing CP-ABE schemes to data access control for cloud storage systems because of the attribute revocation problem.

In this paper, we design an expressive, efficient and revocable data access control scheme for multi-authority cloud storage systems, where there are multiple authorities co-exist and each authority is able to issue attributes independently. Specifically, we propose a revocable multi-authority CP-ABE scheme, and apply it as the underlying techniques to design the data access control scheme. Our attribute revocation method can efficiently achieve both forward security and backward security. The analysis and simulation results show that our proposed data access control scheme is secure in the random oracle model and is more efficient than previous works.

Keywords

Access control, multi-authority, CP-ABE, attribute revocation, cloud storage

I. Introduction

All Data access control is an efficient way to ensure the data security in the cloud. Cloudstorage services allows data owner to outsource their data to the cloud. Attribute-based encryption(ABE) is a new concept of encryption algorithms that allow the encryptor to set a policydescribing who should be able to read the data. In an attribute-based encryption system, private keysdistributed by an authority are associated with sets of attributes and ciphertexts are associated withformulas over attributes. A user should be able to decrypt a ciphertext if and only if their private keyattributes satisfy the formula. In traditional public-key cryptography, a message is encrypted for aspecific receiver using the receiver's public-key. Identity-based cryptography and in particularidentity-based encryption (IBE) changed the conventional understanding of public-key cryptographyby allowing the public-key to be an arbitrary string, e.g., the email address of the receiver. ABE goesone step further and defines the identity not atomic but as a set of attributes, e.g. roles, and messagescan be encrypted with respect to subsets of attributes (key-policy ABE - KP-ABE) or policiesdefined over a set of attributes (ciphertext-policy ABE - CP-ABE).

In ciphertext-policy attribute-based encryption (CP-ABE) a user's private-key is associated with a setof attributes and a ciphertext specifies an access policy over a defined universe of attributes withinthe system. A user will be able to decrypt a ciphertext, if and only if his attributes satisfy the policyof the respective ciphertext. Cipher text-Policy Attribute-based Encryption (CP-ABE) is consideredas one of the most suitable scheme for data access control in cloud storage. This scheme providesdata owners more direct control on access policies. However, CP-ABE schemes to data accesscontrol for cloud storage systems are difficult because of the attribute revocation problem. So Thispaper produce survey on efficient and revocable data access control scheme for multi-authority cloudstorage systems, where there are multiple authorities cooperate and each authority is able to issueattributes independently.

CP-ABE thus allows to realize implicit authorization, i.e., authorization is included into theencrypted data and only people who satisfy the associated policy can decrypt data. Another nicefeature is that users can obtain their private keys after data has been encrypted with respect topolicies. So data can be encrypted without knowledge of the actual set of users that will be able todecrypt, but only specifying the policy which allows decrypting. Any future users that will be given a key with respect to attributes such that the policy can be satisfied will then be able to decrypt thedata.

II. System Model and Security Model

System Model

We consider data access control systeminmulti-authoritycloud storage, as described in Fig. 1. There are five types ofentities in the system: a certificate authority (CA), attributeauthorities (AAs), data owners (owners), the cloud server(server) and data consumers (users).The CA is a global trusted certificate authority in thesystem. It sets up the system and accepts the registration ofall the users and AAs in the system. For each legal user inthe system, the CA assigns a global unique user identity toit and also generates a global public key for this user.

However, the CA is not involved in any attribute managementand the creation of secret keys that are associatedwith attributes. For example, the CA can be the Social Security Administration, an independent agency of theUnited States government. Each user will be issued a SocialSecurity Number (SSN) as its global identity. Every AA is an independent attribute authority that isresponsible for entitling and revoking user's attributesaccording to their role or identity in its domain. In ourscheme, every attribute is associated with a single AA, buteach AA can manage an arbitrary number of attributes.Every AA has full control over the structure and semanticsof its attributes. Each AA is responsible for generating apublic attribute key for each attribute it manages and asecret key for each user reflecting his/her attributes.

Each user has a global identity in the system. A usermaybe entitled a set of attributes which may come frommultiple attribute authorities. The user will receive a secretkey associated with its attributes entitled by thecorresponding attribute authorities.

Each owner first divides the data into several componentsaccording to the logic granularities and encrypts eachdata component with different content keys by usingsymmetric encryption techniques. Then, the owner definesthe access policies over attributes from multiple attributeauthorities and encrypts the content keys under thepolicies. Then, the owner sends the encrypted data to thecloud server together with the ciphertexts.2 They do notrely on the server to do data access control. But, the accesscontrol happens inside the cryptography. That is only when theuser's attributes satisfy the access policy defined in theciphertext, the user is able to decrypt the ciphertext. Thus,users with different attributes can decrypt differentnumber of content keys and thus obtain different granularitiesof information from the same data.

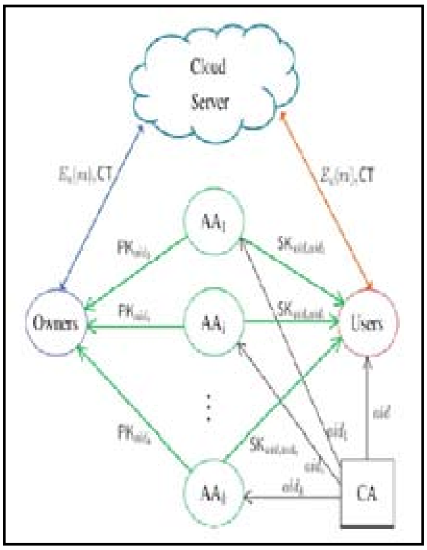

Fig. 1 : System model of data access control in multi-authority cloudstorage.

Security model

In multi-authority cloud storage systems, we make thefollowing assumptions:

- The CA is fully trusted in the system. It will notcollude with any user, but it should be preventedfrom decrypting any ciphertexts by itself.
- Each AA is trusted but can be corrupted by theadversary.
- The server is curious but honest. It is curious aboutthe content of the encrypted data or the receivedmessage, but will execute correctly the task assignedby each attribute authority.
- Each user is dishonest and may collude to obtainunauthorized access to data.

III. CP-ABE

One of the most suitable technologies for data access controlin cloud storage systems is Cipher text-Policy Attribute-basedEncryption (CP-ABE). It provides the data owner to directcontrol on access policies. The Authority in this scheme isresponsible for key distribution and attribute management. Theauthority may be the university Administration office, Staffmaintenance (Human resource-HR) department in a company,etc. The data owner in CP-ABE scheme defines the accesspolicies and encrypts data depending on the policies.

A. CP-ABE Types

In CP-ABE scheme for every user will be issued a secret keyreflecting its attributes. A user can decrypt the data only whenits attributes to satisfy the access policies.

There are two types of CP-ABE systems:

- Single-authority CP-ABE
- Multi-authority CP-ABE

In Single-authority CP-ABE method, where all the attributesare managed by only one a single authority. In a MultiauthorityCP-ABE scheme where attributes are from differentattribute authorities. This method is more suitable for data access control of cloud storage systems. Data users containattributes should be issued by multiple authorities and dataowners. Data users may also share the data using access policydefined over attributes from different authorities.

In our scheme, the data owner does not required totrust the server. Because, the key is based on attribute andmaintained by the attribute authority. We designed newrevocation method for multi-authority CP-ABE. Then, weapply them to design a fully secure and efficient data sharingfor multi-authority scheme.The important advantages of this work can besummarized as follows,

i. We proposed third party auditor (TPA) which usedfor auditing the data.

ii. We develop a new revocation method for userattribute revocation.

B. CP-ABE Alogirthm

A CP-ABE scheme have four algorithms: Setup, Encrypt,KeyGen, and Decrypt.

1. Setup (λ; U)

The setup algorithm takes input as securityparameter and attribute universe description. It outputs theglobal public parameters PK and a global master key MK.

2. Encrypt (PK; M; A)

The encryption algorithm takes as inputthe public parameters PK of attributes, a message M, and anaccess structure A over the involved attributes. The algorithmwill encrypt M and produce a ciphertext (CT) that only a userhaving a set of attributes that satisfies the access structure willbe able to decrypt the message. We will assume that the ciphertext implicitly contains A.

3. Key Generation (MK; S)

The key generation algorithmtakes as input the global master key MK and a set of attributesS that clarify the key. It outputs a private key SK.

4. Decrypt (PK; CT; SK)

The decryption algorithm takes asinput the public parameters PK, a ciphertext (CT), whichcontains an access policy A, and a private key SK, which is aprivate key for a set S of attributes. If the set S of attributessatisfies the access structure A then the algorithm will decryptthe ciphertext and return a message M.

IV. Frame Work

The data access control for Multi-Authority cloud storagesystem consists following methods.

1) System Initialization

- **CA Setup** (1λ): (GMK, GPP, (GPK'uid, GPK'uid), (GSKuid; GSK'uid), Certificate(uid)).
- The CA setup algorithm is run by the CA. Ittakes no input other than the implicit securityparameter λ. It generates the global master key GMKof the system and the global public parameters GPP.For each user uid, it generates the user's global publickeys (GPKuid, GPK'uid), the user's global secret keys(GSKuid ,GSK'uid) and a certificate Certificate (uid) of the user.
- **AASetup (**Uaid):(SKaid, PKaid, {VKxaid, PKxaid } xaid,Uaid).

The attribute authority setup algorithm is runby each attribute authority. It takes the attributeuniverse Uaid managed by the AAaid as input. Itoutputs a secret and public key pair (SKaid, PKaid) ofthe AAaid and a set of version keys and publicattribute keys {VKxaid, PKxaid }xaid,Uaid for all theattributes managed by the AAaid.

2) Attribute Authority's key generation and management

Secret Key Distribution

A randomized algorithmtakes as input the authority's secret key SK, a user u'sUID, and a set of attributes Aku in the authority AAk'sdomain (We will assume that the user's claim of theseattributes has been verified before this algorithm isrun, Au = {Aku , k = 1, . . . , n}). Output a secret keyDu for the user u.

Access issue id Distribution

The collected attributesfrom all attribute authorities (Aa) will be sent to theusers for the encryption purpose.

3) Data Encryption

The data owner runs the encryption algorithm to encrypt thecontent keys. By using symmetric encryption method the datais encrypted with content keys. A randomized algorithm takesas input a set of public key of attributes involved in encryption, a message M, the global public parameters GPPand outputs the ciphertext C.

4) Data Decryption

The users first run the decryption algorithm and use them toVdecrypt data's from the ciphertext C. It takes input theVciphertext C, it have access policy with itself for verifying theVaccess rules of the users. If the access policy is satisfied withVthe users attribute, the decryption algorithm will decrypt theVciphertext C.

5) Attribute revocation:

The attribute revocation has been solved by assigning newversion key VK for non-revoked attribute. It takes as inputsthe secret key of Attribute authority, revoked attribute id and
current version key. Its outputs as new version key and newattribute key.

V. Our Data Access Control Scheme

In this section, we first give an overview of the challengesand techniques. Then, we propose the detailed constructionof our access control scheme which consists of fivephases: System Initialization, Key Generation, Data Encryption,Data Decryption and Attribute Revocation.

To design the data access control scheme for multiauthority cloud storage systems, the main challengingissue is to construct the underlying Revocable Multiauthority CP-ABE protocol. In, Chase proposed amulti-authority CP-ABE protocol, however, it cannot be directly applied as the underlying techniques because oftwo main reasons: 1) Security Issue: Chase's multi-authorityCP-ABE protocol allows the central authority to decrypt allthe ciphertexts, since it holds the master key of the system;2) Revocation Issue: Chase's protocol does not supportattribute revocation.

We propose a new revocable multi-authority CP-ABEprotocol based on the single-authority CP-ABE proposedby Lewko and Waters in. That is we extend it to multiauthorityscenario and make it revocable. We apply thetechniques in Chase's multi-authority CP-ABE protocolto tie together the secret keys generated by differentauthorities for the same user and prevent the collusionattack. Specifically, we separate the functionality of theauthority into a global certificate authority (CA) andmultiple attribute authorities (AAs). The CA sets up thesystem and accepts the registration of users and AAs in thesystem. It assigns a global user identity uid to each userand a global authority identity aid to each attribute authorityin the system. Because the uid is globally unique inthe system, secret keys issued by different AAs for the sameuid can be tied together for decryption. Also, because eachAA is associated with an aid, every attribute is distinguishableeven though some AAs may issue the same attribute.

To deal with the security issue in, instead of using thesystem unique public key (generated by the unique masterkey) to encrypt data, our scheme requires all attributeauthorities to generate their own public keys and uses themto encrypt data together with the global public parameters.This prevent the certificate authority in our scheme fromdecrypting the ciphertexts.

To solve the attribute revocation problem, we assign aversion number for each attribute. When an attributerevocation happens, only those components associatedwith the revoked attribute in secret keys and ciphertextsneed to be updated. When an attribute of a user is revoked
from its corresponding AA, the AA generates a new versionkey for this revoked attribute and generates an update key.

With the update key, all the users, except the revoked user,who hold the revoked attributes can update its secret key(Backward Security). By using the update key, thecomponents associated with the revoked attribute in theciphertext can also be updated to the current version. Toimprove the efficiency, we delegate the workload ofciphertext update to the server by using the proxy reencryptionmethod, such that the newly joined user is alsoable to decrypt the previously published data, which areencrypted with the previous public keys, if they havesufficient attributes (Forward Security). Moreover, byupdating the ciphertexts, all the users need to hold onlythe latest secret key, rather than to keep records on all theprevious secret keys.

VI. Security Analysis

We prove that our data access control is secure under the security model we defined, which can be summarized as in the following theorems.

Theorem 1. When the decisional q-parallel BDHE assumption holds, no polynomial time
adversary can selectively break our system with a challenge matrix of size l_ n_, where n_ _ q.

Proof. The proof is given in the supplemental file available online. g

Theorem 2. Our scheme can achieve both Forward Security and Backward Security.

ISSN : 2394-2975 (Online)
ISSN : 2394-6814 (Print)

Backward Security: During the secret key update phase, the corresponding AA generates an update key for each non-revoked user. Because the update key is associated with the user's global identity uid, the revoked user cannot use update keys of other non-revoked users to update its own secret key, even if it can compromise some non-revoked users. Moreover, suppose the revoked user can corrupt some other AAs (not the AA corresponding to the revoked at-tributes), the item HðxaidÞvxaid _aid_aid in the secret key can prevent users from updating their secret keys with update keys of other users, since _aid is only known by the AAaid and kept secret to all the users. This guarantees the back-ward security.

Forward Security: After each attribute revocation oper-ation, the version of the revoked attribute will be updated. When new users join the system, their secret keys are as-sociated with attributes with the latest version. However, previously published cipher texts are encrypted under at-tributes with old version. The cipher text update algorithm in our protocol can update previously published cipher-texts into the latest attribute version, such that newly joined users can still decrypt previously published cipher texts, if their attributes can satisfy access policies associated with cipher texts. This guarantees the forward security.

Theorem 3. Our access control scheme can resist the collusion attack, even when some AAs are corrupted by the adversary. each other, although some AAs may issue the same attributes. Moreover, the secret key is also associated with the user's globally unique identity uid. Thus, users cannot collude together to gain illegal access by combining their attributes together.

However, when some AAs is corrupted by the adver-sary, the collusion resistance becomes more complicated. Specifically, the adversary may launch Attribute Forge Attack, defined as follows. Suppose a user uid0 possesses an attribute ‘‘xaid0 ’’ from AAaid0 , while the adversary does not hold the attribute ‘‘xaid0 ’’ from AAaid0 . The adversary attempts to forge (‘‘clone’’) the attribute ‘‘xaid0 ’’ from the user uid0's secret key by colluding with some other AAs. In our scheme, the item gu0uid tuid;aid _aid in the secret key construction helps to resist this attack. When the adversary corrupts any AAs, he/she can get all the global secret key GSKuid for all the users in the system (because each AA has full knowledge on one of the user's global secret keys GSKuid). Suppose all the Kxaid ;uid in the secret key is constructed without this item. The adversary can success- fully forge the attribute ‘‘xaid0 ’’ as Privacy-Preserving Guarantee: Although the CA holds the global master key GMK, it does not have any secret key issued from the AA. Without the knowledge of g_aid , the CA cannot decrypt any ciphertexts in the system. Our scheme can also prevent the server from getting the content of the cloud data by using the proxy-encryption method.

VII. Performance Analysis

In this section, we analyze the performance of our scheme by comparing with the Ruj's DACC scheme and our previous scheme in the conference version, in terms of storage overhead, communication cost and computation efficiency. We conduct the comparison under the same security level. Let jpj be the element size in the G; GT ; Zp. Suppose there are nA authorities in the system and each attribute authority AAaid manages naid attributes. Let nU and nO be the total number of users and owners in the system respectively. For a user uid, let nuid;aidk ¼ jSuid;aidk j denote the number of attributes that the user uid obtained from AAaidk. Let ‘ be the total number of attributes in the ciphertext.

VIII. Feature Work

Ciphertext-Policy Attribute-based Encryption (CP-ABE), is regarded as one of the most suitable technologies for data access control in cloud storage systems, because it gives the data owner more direct control on access policies. In CP-ABE scheme, there is an authority that is responsible for attribute management and key distribution. The authority can be the registration office in a university, the human resource department in a company, etc. The data owner defines the access policies and encrypts data according to the policies. Each user will be issued a secret key reflecting its attributes. A user can decrypt the data only when its attributes satisfy the access policies. In this paper, we first propose a revocable multi authority CP-ABE scheme, where an efficient and secure revocation method is proposed to solve the attribute revocation problem in the system.

1. We modify the framework of the scheme and make it more practical to cloud storage systems, in which data owners are not involved in the key generation.
2. We greatly improve the efficiency of the attributer vocation method.

IX. Conclusion

In this paper, we proposed a revocable multi-authority CPABE scheme that can support efficient attribute revocation. Then, we constructed an effective data access control scheme for multi-authority cloud storage systems. We also proved that our scheme was provable secure in the random oracle model. The revocable multi-authority CPABE is a promising technique, which can be applied in any remote storage systems and online social networks etc.

Ciphertext-Policy Attribute-Based Encryption (CP-ABE) is a promising technique that is designed for access control of encrypted data. There are two types of CP-ABE systems: single authority CP-ABE where all attributes are managed by a single authority, and multi-authority CP-ABE, where attributes are from different domains and managed by different authorities. Multi-authority CP-ABE is more appropriate for the access control of cloud storage systems.

References

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

[2] *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.*

[3] *A.B. Lewko and B. Waters, "Decentralizing Attribute-Based Encryption," in Proc. Advances in Cryptology-EUROCRYPT'11, 2011, pp. 568-588.*

[4] *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 (ASIACCS'10), 2010, pp. 261-270.*

[5] *M. Li, S. Yu, Y. Zheng, K. Ren, andW. 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.*

[6] *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.

[7] S. Jahid, P. Mittal, and N. Borisov, "Easier: Encryption-Based Access Control in Social Networks with Efficient Revocation," in Proc. 6th ACM Symp. Information, Computer and Comm. Security (ASIACCS'11), 2011, pp. 411-415.

[8] 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.

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

[10] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," in Proc. 21st Ann. Int'l Cryptology Conf.: Advances in Cryptology - CRYPTO'01, 2001, pp. 213-229.

[11] A.B. Lewko and B. Waters, "New Proof Methods for Attribute-Based Encryption: Achieving Full Security through Selective Techniques," in Proc. 32st Ann. Int'l Cryptology Conf.: Advances in Cryptology - CRYPTO'12, 2012, pp. 180-198.

[12] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," National Institute of Standards and Technology, Gaithersburg, MD, USA, Tech. Rep., 2009.