ISSN 2348-375X

Unique Journal of Engineering and Advanced Sciences

Available online: www.ujconline.net

Research Article

DECENTRALIZED ACCESS CONTROL MECHANISM FOR CLOUD BASED DATA STORAGE

Vinoth Kumar J[1*], Vikramarajan Jambulingam[2]

[1]Department of Computer Science and Engineering, Bhajarang Engineering College, Tamil nadu, India
[2]Department of Electrical and Computer engineering, University of Gondar

Received: 27-04-2015; Revised: 25-05-2015; Accepted: 22-06-2015

*Corresponding Author: **J. Vinoth Kumar**
Department of Computer Science and Engineering, Bhajarang Engineering College, Tamil nadu, India, Mobile: 09677823604

ABSTRACT

We propose a novel decentralized access control mechanism for safer data storage in clouds that ropes anonymous authentication. In the proposed mechanism, the cloud check is used to authenticate, the series without knowing the user's identity before actually storing data. Our mechanism also has the additional feature of access control in which only legitimate users are able to decrypt the stored information. The mechanism prevents replay attacks and provides the abilities such as creation, modification, and reading data stored in the cloud. We also address user revocation. Also, our access control and authentication mechanism is decentralized and secure, unlike other access control mechanisms developed for clouds which are mostly centralized. The communication, calculation, and storage overheads are better when compared to the centralized approaches.

Keywords: Decentralized access control, Anonymous authentication, Encryption, Cloud based storage.

INTRODUCTION

In cloud computing, users can farm out their calculation and storage to servers through Internet. This frees users from the problems of maintaining resources on the place. Clouds can give many types of services such as (Google Apps, Microsoft online and infrastructures such as Amazon's EC_2, and platforms to assist developers create applications such as Amazon's s_3. Most of the data saved in clouds is highly susceptible, say medical records and social networking data. Security and privacy becomes very important issues in cloud based environment. So, the users must authenticate themselves before initiating any transaction, and also, it must be ensured that the cloud does not interfere with the data that is outsourced. User privacy is also needed so that the cloud users do not identify the identity of the user.

Cloud servers are vulnerable to Byzantine failure where a storage server may fail in random ways[1]. Proficient search on encrypted data is also a necessary concern in clouds. The clouds must not identify the query but must be able to return the records that satisfy the query. This is performed by ways of searchable encryption[2-3]. Authentication of users using public key crypto-graphic techniques had been discussed[4]. Many homomorphic encryption techniques had been studied to ensure that the cloud's is not able to read the data while doing calculations on them[5-6].

Access control schemes in clouds are gaining awareness since it is important that only allowed users have access to legitimate service. Care must be taken to ensure access control of this susceptible information which can mostly be related to health, documents as in Google doc. (or) even personal information such as social networking. There are basically three types of access control: user-based access control (UBAC), role-based access control (RBAC), and attribute-based access control (ABAC). In UBAC, the access control list consists of list of users who are allowed to access data. There are cryptographic schemes like ring signatures[13], mesh signatures[14], group signatures[15], which can be useful in these situations. Ring signatures are not a realistic option for clouds where there are a huge number of users. Group signatures consider the pre-existence of a group which might not be possible in clouds. Mesh signatures do not make certain if the message is from a single user or multiple users colluding together. Previous works[7,8,10-12], on access control in cloud is centralized in nature. The scheme in uses a symmetric key technique and does not support authentication. The schemes[7,8,11] do not support authentication as well.

The main goals of this paper are the following:

1. Distributed access control of data stored in cloud so that only allowed users with legitimate attributes can access them.
2. Authentication of cloud users who store and change their data on the cloud.
3. The identity of the user is confined from the cloud during authentication.
4. The architecture is decentralized meaning that there can be multiple KDCs for key management.
5. The access control and authentication are both collusion opposing, meaning that no two users can get together and access data or authenticate themselves, if they are individually not allowed.
6. Canceled users cannot access data after they have been canceled.
7. The proposed scheme is flexible to replay attacks. A writer whose attributes and keys have been canceled cannot write back state information.

Organization

The paper is structured as follows: Related work is detailed in Part 2. The mathematical backgrounds are detailed in Part 3. We detail our privacy preserving access control scheme in Part 4. And comparisons with other works are detailed in Part 5. We conclude in Part 6.

RELATED WORK

There are two types of ABEs. In key-policy ABE (Goyal et al.) [17], the sender has an access policy to encrypt data. A writer whose attributes and keys have been canceled shall not write back stale information. The receiver gets attributes and secret keys from the attribute authority and is able to decrypt information if it has corresponding attributes. In Cipher text policy CP-ABE[18], the receiver has the access policy in the form of a tree, with attributes as leaves and monotonic access structure with AND, OR and other threshold gates. All the approaches take a centralized approach and can allow only one KDC, which is a point of failure.

BACKGROUND

In this part, we detail our cloud storage model, attack model and the assumptions we have made in this work. Table 1 details the notations used throughout the paper. We also describe mathematical background used in our proposed solution.

Assumptions

We make the following assumptions in our work:

The cloud is honest-but-curious, which means that the cloud administrators can be interested in checking user's content, but cannot change it.

1. Users can have either read or write or done both accesses to a file stored in the cloud.
2. All communications between users of clouds are protected by secure shell protocol.

Formats of Access Policies

Access policies can be in any of the following formats: 1) Boolean functions of attributes, 2) linear secret sharing scheme matrix, or 3) monotone span programs. Any access structure can be converted into a Boolean function as well.

Symbols	Meanings
U_u	u-th User/Owner
$\mathcal{A}_j$	j-th KDC
$\mathbb{A}$	Set of KDCs
L_j	Set of attributes that KDC $\mathcal{A}_j$ possesses
$l_j = \|L_j\|$	Number of attributes that KDC $\mathcal{A}_j$ possesses
$I[j,u]$	Set of attributes that $\mathcal{A}_j$ gives to user U_u for encryption/decryption
I_u	Set of attributes that user U_u possesses
$J[j,u]$	Set of attributes that $\mathcal{A}_j$ gives to user U_u for claim attributes
J_u	Set of attributes that user U_u possesses as claim attributes
$AT[j]$	KDC which has attribute j
$PK[j]/SK[j]$	Public key/secret key of KDC $\mathcal{A}_j$ for encryption/decryption
$sk_{i,u}$	Secret key given by $\mathcal{A}_j$ corresponding to attribute i given to user U_u
TPK/PSK	Trustee public key/secret key
$APK[j]/ASK[j]$	Public key/secret key of KDC $\mathcal{A}_j$ for verifying claim
$\mathcal{X}$	Boolean access structure
$\mathcal{Y}$	Claim policy
τ	Time instant
R	Access matrix of dimension $m \times h$
M	Matrix of dimension $l \times t$ corresponding to the claim predicate
MSG	Message
$\|MSG\|$	Size of message MSG
C	Ciphertext
$H, \mathcal{H}$	Hash functions, example SHA-1

Figure 1: Notations used

Mathematical Background

Let G be a cyclic group of prime order q generated by g. Let G_T be a group of order q. We can define the map e: G*G-->G_T. The map satisfies the following properties:

1. $e(aP,bQ) = e(P,Q)^{ab}$ for all P,Q ∈ G
   and $a,b \in Z_q = \{0,1,2,3\ldots,q-1\}$
2. Nondegenarate: $e(g,g) \neq 1$.

Bilinear pairing on elliptic curves groups is used. We will not discuss the pairing functions which mainly use Weil and Tate pairings and computed using Miller's scheme. The choice of curve is an important consideration because it chooses the complexity of pairing operations.

Attribute-Based Encryption

ABE with multiple authorities as proposed by Lewko and Waters, proceeds as follows[11]:

System Initialization

Select a prime q, generator g of G_0, groups G_0 and G_T of order q, a map e: G_0* G_0→G_T, and a hash function H: {0, 1}→ G_0 that maps the identities of users to G_0.

The set of attributes L_j..The Secret key of KDC is SK[j]={αi,yi,i∈Lj} and the public key of KDC is given as PK[j]={$e(g,g)^{\alpha}$,g^{y},i∈Lj}.

Key Generation and Distribution by KDCs

User U_u receives a set of attributes I[j,u] from KDC and Corresponding secret key $SK_{i,u}$ for each i ∈ I[j,u]

Ski,u = g^{α} $H(u)^{y}$. All keys are delivered to the user using public key of user such that only that particular user may decrypt it using its secret key.

Encryption by Sender

The encryption function is ABE. Encrypt (MSG, X). Sender decides about the access tree X. LSSS matrix R can be derived as described earlier. Sender encrypts message MSG and sends corresponding cipher text.

Decryption by Receiver

The decryption function is ABE Decrypt (C{$sk_{i,u}$}) where C is the cipher text. Receiver U_u takes as input ciphertext C, secret $sk_{i,u}$, group G_o and outputs message msg.

Proposed privacy preserving access control mechanism

In this part, we propose our privacy preserving access control scheme. According to this scheme a user can create a file and store it safely in the cloud. This scheme consists of two protocols ABE and ABS, as discussed in previous sections correspondingly. We will first discuss our scheme and then demonstrate how it works. We refer to the Fig. 1. There are three users, a creator, a reader, and writer. Creator Alice gets a token from the trustee, who is thought to be honest. A trustee can be someone like the central government who manages social security numbers etc. On detailing her id, the trustee provides her token. There are multiple KDCs, which can be spread. For example, these can be servers in different areas of the world. A creator on detailing the token to one or more KDCs gets keys for encryption/decryption and signing. In the Fig.1, SKs are secret keys given for decryption, K_x are keys for signing purpose. The message named as MSG is encrypted under the access policy of X. The access policy decides who can and cannot access the data stored in the cloud. The creator decides on a claim policy of Y, to prove her authenticity and signs the message under this claim. The cipher text C with signature is termed as c, and is sent to the cloud. The cloud verifies the signature.

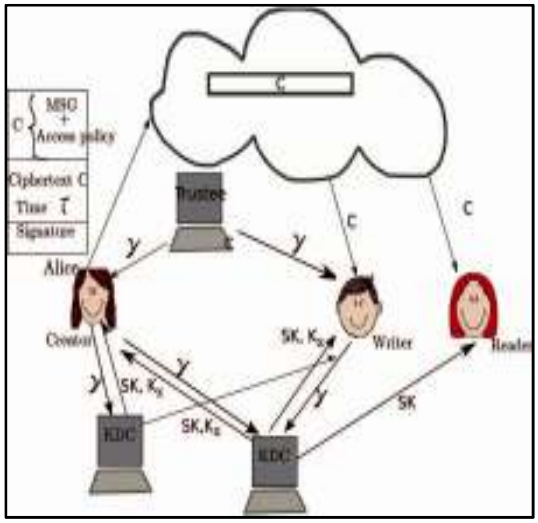

Figure 2: Secure cloud storage model

Data Storage in Clouds

A user U_u first registers itself with one or more trustees. For simplicity we assume there is one trustee. The user then creates an access policy X which is a monotone Boolean function. The message is then encrypted under the access policy as with encryption being ABE.The user also constructs a claim policy of Y to enable the cloud to authenticate the user. The creator does not send the message MSG as it is, but uses the time stamp and creates H(C). This is done to prevent replay attacks. If the time stamp is not sent, then the user can write previous stale message back again to the cloud with a valid signature, even when its claim policy and attributes have been canceled. The original work by Maji et al., [16]suffers from replay attacks. In their scheme, a writer can send its message and correct signature even when there is no longer has access rights. In our scheme a writer whose rights have been canceled cannot create a new signature with new time stamp and, hence cannot write back stale information.

Reading from the Cloud

When a user requests data from the cloud, the cloud sends the cipher text C using SSH protocol. Decryption proceeds using algorithm ABE (Decrypt) and the message MSG is calculated.

Writing to the Cloud

To write to an already existing file, the user must send its message with the claim policy as done during file creation. The cloud verifies the claim policy, and only if the user is authentic, is allowed to write on the file.

User Revocation

We have just discussed how to prevent replay attacks. We will now discuss how to handle user revocation. It must be ensured that users must not have the ability to access data, even if they possess matching set of attributes. For this reason, the owners must change the stored data and send updated information to other users. The set of attributes I_u possessed by the Canceled user U_u is noted and features that the other schemes did not support.

Comparison with other Cloud Based Access Control Mechanisms

1-W-M-R means that only one user can write while many users can read. M-W-M-R means that many users can write and read. We see that most schemes do not support many writes which is supported by our scheme. Our scheme is robust and decentralized; most of the others are centralized. Our scheme also supports privacy preserving authentication, which is not supported by others. Most of the schemes do not support user revocation, which our scheme does. We compare the calculation and communication costs incurred by the users and clouds and show that our distributed approach has comparable costs to centralized approaches. The most expensive operations involving pairings and is done by the cloud. If we compare the calculation load of user during read we see that our scheme has comparable costs. Our scheme also compares well with the other authenticated scheme.

CONCLUSION

We have provided a decentralized access control technique with anonymous authentication, which provides user cancelation and prevents threats like replay attacks. The cloud does not know the identity of the user who saves information, but only verifies the user's credentials. Key distribution is done in a decentralized manner. One limitation is that the cloud knows the access policy for every record stored in the cloud. In future, we would like not to show the attributes and access policy of a user.

REFERENCES

1. Wang C, Wang Q, Ren K, Cao N, and Lou W, Toward Secure and Dependable Storage Services in Cloud Computing, IEEE Trans. Services Computing, 2012; 5(2): 220-232.

2. Li J, Wang Q, Wang C, Cao N, Ren K, and Lou W, Fuzzy Keyword Search Over Encrypted Data in Cloud Computing, Proc. IEEE INFOCOM, 2010; 441-445.
3. Kamara S and Lauter K, Cryptographic Cloud Storage, Proc. 14th Int'l Conf. Financial Cryptography and Data Security, 2010: 136-149.
4. Li H, Dai Y, Tian L, and Yang H, Identity-Based Authentica-tion for Cloud Computing, Proc. First Int'l Conf. Cloud Computing (CloudCom), 2009:157-166.
5. Gentry C, A Fully Homomorphic Encryption Scheme, PhD dissertation, Stanford Univ., 2009.
6. Sadeghi A R, Schneider T, and Winandy W, Token-Based Cloud Computing, Proc. Third Int'l Conf. Trust and Trustworthy Computing (TRUST), 2010:417-429.
7. Li M, Yu S, Ren K, and Lou W, Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-Owner Settings, Proc. Sixth Int'l ICST Conf. Security and Privacy in Comm. Networks 2010: 89-106.
8. Yu S, Wang C, Ren K, and LouW, Attribute Based Data Sharing with Attribute Revocation, Proc. ACM Symp. Information, Computer and Comm. Security, 2010:261-270.
9. Wang G, Liu Q, and Wu J, Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services, Proc. 17th ACM Conf. Computer and Comm. Security, 2010:735-737.
10. Zhao F, Nishide T, and Sakurai K, Realizing Fine-Grained and Flexible Access Control to Outsourced Data with Attribute-Based Cryptosystems, Proc. Seventh Int'l Conf. Information Security Practice and Experience (ISPEC), 2011:83-97.
11. Ruj S, Nayak A, and Stojmenovic I, DACC: Distributed Access Control in Clouds, Proc. IEEE 10th Int'l Conf. Trust, Security and Privacy in Computing and Communications, 2011.
12. Jahid S, Mittal P, and Borisov N, EASiER: Encryption-Based Access Control in Social Networks with Efficient Revocation, Proc. ACM Symp. Information, Computer and Comm. Security, 2011.
13. Rivest RL, Shamir A, and Tauman Y, How to Leak a Secret, Proc. Seventh Int'l Conf. Theory and Application of Cryptology and Information Security, 2001:552-565.
14. Boyen X, Mesh Signatures, Proc. 26th Ann. Int'l Conf. Advances in Cryptology , 2007:210-227.
15. Chaum D and Heyst E V, Group Signatures, Proc. Ann. Int'l Conf. Advances in Cryptology, 1991: 257-265.
16. Maji HK, Prabhakaran M, and Rosulek M, Attribute-Based Signatures: Achieving Attribute-Privacy and Collusion-Resis-tance, IACR Cryptology ePrint Archive, 2008.
17. Goyal V, Pandey O, Sahai A, andWaters B, Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data, Proc. ACM Conf. Computer and Comm. Security, 2006:89-98.
18. Bethencourt J, Sahai A, and Waters B, Ciphertext-Policy Attribute-Based Encryption," Proc. IEEE Symp. Security and Privacy, 2007:321-334.
19. Liang X, Cao Z, Lin H, and Xing D, Provably Secure and Efficient Bounded Cipher text Policy Attribute Based Encryption, Proc. ACM Symp. Information, Computer and Comm. Security, 2009:343-352.

J. Vinothkumar received his Master and Bachelor degree in Computer Science Engineering from affiliated colleges of Anna University, India. His research interests are computer networks, network security and cloud computing.

Vikramarajan Jambulingam received his Master degree in Power Electronics and Drives and Bachelor degree in Electrical and Electronics Engineering from VIT University, India. His research interests are power electronic applications, power quality, power electronic converters and computer networks.

Source of support: Nil, Conflict of interest: None Declared