International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Secure Cloud Storage with Anonymous Authentication using Decentralized Access Control

Jessy Cherian

Mount Zion College of Engineering, Kadammanitta, Pathanamthitta

Abstract: ***Security and privacy protection in clouds are challenging. Efficient search in clouds is also an important concern. A new decentralized access control scheme that supports anonymous authentication for secure data storage is proposed. In this scheme without knowing the user's identity before storing the data the cloud verifies the authenticity. This scheme also helps in creation, modification and reading data stored in cloud and also prevents replay attacks.***

Keywords: Access control, authentication, attribute-based encryption, cloud storage.

1. Introduction

Cloud computing is receiving a lot of challenges in both academic and industrial fields. Here users outsource their storage and computation to servers called clouds by the use of Internet. This frees users from the hardness of managing resources on-site. Certain services of cloud like applications (e.g., Google Apps, Microsoft online), infrastructures (e.g., Amazon's EC2, Eucalyptus) and platforms that help developers to write applications.

Most of the data stored in cloud is sensitive like medical records and social networks. Thus, security and privacy are two major issues in cloud computing. In one hand user must authenticate itself before any transaction and on other hand cloud does not interfere with the data outsourced.

Efficient search on encrypted data is also an important issue in clouds. The query should not be known by the cloud but should be able to return the records that satisfy the query. These can be achieved by various searchable encryption methods [1] and [2]. Here the keywords sent to the cloud are encrypted and returns the results without knowing the actual keyword for the search. The problem here is that the records should have the keywords associated with them to enable search. Exact records are returned only when searched with correct keywords.

Access control is important in because only authorized users can access data. Another important fact the information must be from a reliable source. Since large data stored in cloud is highly sensitive, care must be taken to ensure access control of this sensitive information. Access control is also gaining much attention in online social networking sites where users use their personal information like pictures, videos, audios and documents.

Anonymity of the user is also necessary, if the user doesn't need to disclose his/her identity. But the user need to prove that the information stored is from a genuine user. Apart from various cryptographic protocols like ring signatures, mesh signatures[3], group signatures [4], a new protocol called Attribute- Based signature (ABS) has been applied [4].

Previous work [5], [6], [7], [8], [9], [10], [11] on access control is centralized. Since authors take a centralized approach that is single key distribution center (KDC) that distributes secret keys and attributes to all users. In this paper, a new decentralized approach is used. In ABS scheme is also used to achieve privacy and authenticity. This scheme is resistant to replay attacks, in which a user can replace fresh data with old information from previous write.

2. Related Work

A protocol called Attribute-Based Encryption (ABE) [7] is used that contain a set of attributes in addition to its unique ID. Two classes of ABE are key-policy ABE and Ciphertext-policy ABE. In key-policy the sender has an access policy to encrypt data. In Ciphertext-policy receiver has the access policy in the form of a tree in which attributes are leaves and have access structures with AND, OR and other gates.All the approaches takes a centralized format and allow only one KDC with a single point failure.

3. Background

In this section, a cloud storage model and the assumptions that have been made in this paper. Table 1 consists of certain notations used in this paper

3.1 Assumptions

The following assumptions are made in this paper:

1. The cloud is honest-but-curious, cloud administrators can view the contents but cannot update it. This is a correct assumptions [5] and [6].
2. Users can read or write or both accesses to a file stored in the cloud.
3. All the communications between the users in cloud are secured by secure shell protocol (SSH).

3.2 Format of Access Policies

The access policies can be 1) Boolean functions of attributes 2) linear secret sharing scheme (LSSS) matrix, 3) monotone span programs. An example of a Boolean function is $((b1 \wedge b2 \wedge b3) \vee (b4 \wedge b5)) \wedge (b6 \vee b7))$, where a1, a2,…. are attributes.

Table 1: Some Notations

Symbols	Meanings
U_u	u-th user/ owner
Aj	j-th KDC
U	Set of KDCs
W	Set of attributes
ω = |W|	Number of attributes
H_j	Set of attributes that KDC Aj possesses
h_j =| H_j |	Number of attributes that KDC Aj possesses
I[j,u]	Set of attributes that Aj gives to user U_u
N_i	Set of attributes that user ni possesses
PK[j]	Public key of KDC Aj
SK[j]	Secret key of KDC Aj
sk_i, u	Secret key given by Aj corresponding to attribute i given to user U_u
B	Boolean access structure
D	Access matrix of dimension n × h
G	Order of group G
M	Message
CT	Ciphertext
HF	Hash function, example SHA-1

4. Proposed Work

Data stored in cloud follows a distributed access control scheme such that only authorized users with valid attributes can access the data. Authentication of users performs modification and storage of the data in the cloud. At the time of authentication users identity is protected from the cloud. The cloud storage model is decentralized such that there can be several KDCs for key management. Access control and authentication schemes are both collusion resistant. Collusion resistant is that no two users can collude and access data or authenticate themselves, eventhough they are not individually authorized. Revoked users cannot access data after she/he have been revoked.

The scheme proposed is flexible to replay attacks. The proposed protocol also supports multiple read and writes on the data stored in the cloud. The costs of decentralized approach should be less comparable to the existing centralized approaches.

4.1 System Architecture

The proposed architecture is shown in the Fig 1. There are three users, a creator, a reader, and writer. Creator Alice receives a token _ from the trustee, who is assumed to be honest. A trustee can be someone like the federal government who manages social insurance numbers etc. On presenting her id (like health/social insurance number), the trustee gives her a token _. There are multiple KDCs (here 2), which can bescattered. For example, these can be servers in different parts of the world. A creator on presenting the token to one or more KDCs receives keys for encryption/decryption and signing. In the Fig. 1, SKs are secret keys given for decryption, K_x are keys for signing. The message MSG is encrypted under the access policy X. The access policy decides who can access the data stored in the cloud. The creator decides on a claim policy Y, to prove her authenticity and signs the message under this claim. The cipher text C with signature is c, and is sent to the cloud. The cloud verifies the signature and stores the cipher text C. When a reader wants to read, the cloud sends C. If the user has attributes matching with access policy, it can decrypt and get back original message.

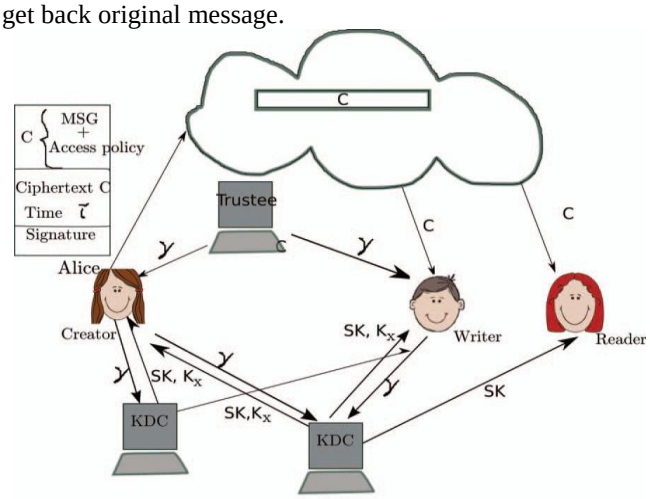

Figure 1: Privacy Preserving Secure Cloud Storage

4.2 Mathematical Background

There are mainly four steps: 1) SystemInitialization, 2) Key and attribute distribution to users By KDCs 3) Encryption of message by sender 4) Decryption by receiver.

System Initialization: Select a prime p, generator r of G_1, groups G_1 and G_2 of order p, m a map : $G_1 \times G_1 \rightarrow G_2$, and a hash function HF : $\{0,1\}^* \rightarrow G_1$which maps the identities of users to G_1. The hash function used here is SHA-1. Each KDC Aj ∈ U has a set of attributesH_j. The attributes disjoint (Hi∩H_j = φ for i ≠ j). Each KDC also chooses tworandom exponents $\sigma_i, \omega_i \in Z_p$. The secret key of KDC Aj is

$$SK[j] = \{\sigma_i, \omega_i, i \in H_j\} \quad (1)$$

The public key of KDC A_j is

$$PK[j] = \{m(r,r)^{\sigma i}, r^{\omega i}, i \in H_j\} \quad (2)$$

2) Key generation and distribution by KDCs: User U_u receives a set of attributes I[j, u] from KDC Aj, and corresponding secret key $sk_{i,u}$ for each i ∈ I[j, u],

$$sk_{i,u} = r^{\sigma i} HF(u)^{\omega i} \quad (3)$$

whereσ_i, ω_i∈SK[j]. Note that all keys are delivered to the user securely using the user's public key, such that only that user can decrypt it using its secret key.

Encryption by sender: Sender decides about the access tree. Sender encrypts message M as follows*:*

1. Choose a random seed s $\in Z_p$and a random vector v $\in Z_p^h$, with s as its first entry; h is the number ofleaves in the access tree.
2. Calculate $\beta_x = R_x \cdot$ v, where R_x is a row of R.
3. Choose a random vector $\delta \in Z_p^h$ with 0 as the first entry
4. Calculate $\delta_x = R_x \cdot \delta$.
5. For each row R_x of R, choose a random $\rho_x \in Z_p$.
6. The following parameters are calculated:

C_0 = M m$(r,r)^s$

$$C_{1,x} = m(r,r)^{\delta x} \; m\,(r,r)^{\sigma\pi(x)\rho x}, \forall x$$

$$C_{2,x} = r^{\rho x} \; \forall x \quad (4)$$

$C_{3,x} = r^{\omega\pi\ (x)\rho x}\, r^{\delta x}\, \forall x$

where π(x) is mapping from R_x to the attribute i thatis located at the corresponding leaf of the access tree.

7. The ciphertext C is sent by the sender:

$$C = R, \pi, C_0, \{\ C_{1,x}\ ,\ C_{2,x}\ , C_{3,x}\ , \}\forall x \qquad (5)$$

Decryption by receiver: Receiver U_u takes as input ciphertext CT, secret keys $\{sk_{i,u}\}$, group G_1, and outputs message M. It obtains the access matrix R and mapping π from CT. It then executes the following steps:

1) Calculates the set of attributes $\{\pi(x): x \in X\} \cap I_u$ that are common to itself and the access matrix. X is the set of rows of R.
2) For each of these attributes, it checks if there is a subset X' of rows of R, such that the vector (1,0.,0)is their linear combination. If not, decryption is impossible.
3) Decryption proceeds as follows:
   a. For each x $dec(x) = \frac{C_{1,x}\ e(H(u), C_{3,x}}{e\ (sk_{\pi}(x), u, C_{2,x})}$)
   b. U_u computes M = $C_0 / \Pi_{x \in X'}\ dec(x)$.

References

[1] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy Keyword Search Over Encrypted Data in Cloud Computing," Proc. IEEE INFOCOM, pp. 441-445, 2010.)
[2] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc. 14th Int'l Conf. Financial Cryptography and Data Security, pp. 136-149, 2010.
[3] X. Boyen, "Mesh Signatures," Proc. 26th Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 210-227, 2007.
[4] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-Based Signatures: Achieving Attribute-Privacy and Collusion-Resistance," IACR Cryptology ePrint Archive, 2008.
[5] M. Li, S. Yu, K. Ren, and W. Lou, "Securing Personal Health Records in Cloud Computing: Patient-Centricand Fine-Grained Data Access Control in Multi-OwnerSettings," Proc. Sixth Int' ICST Conf. Security and Privacy in Comm. Networks (SecureComm), pp. 89-106, 2010
[6] [6] S. Yu, C. Wang, K. Ren, and W. Lou, "AttributeBasedDataSharingwithAttributeRevocation," Proc. ACM Symp. Information, Computer and Comm.Security (ASIACCS), pp. 261-270, 2010.
[7] [7] G. Wang, Q. Liu, and J. Wu, "Hierarchical Attribute Based Encryption for Fine-Grained Access Control in Cloud Storage Services," Proc. 17th ACM Conf. Computer and Comm. Security (CCS), pp. 735-737, 2010.
[8] [8] F. Zhao, T. Nishide, and K. Sakurai, "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), pp. 83-97, 2011.
[9] [9] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed AccessControl in Clouds," Proc. IEEE 10th Int'l Conf. Trust, Security andPrivacy in Computing and Communications TrustCom), 2011
[10] [10] http://securesoftwaredev.com/2012/08/20/xacml-in-the-cloud, 2013.
[11] [11] W. Wang, Z. Li, R. Owens, and B. Bhargava, "Secure and EfficientAccess to Outsourced Data," Proc. ACM Cloud Computing SecurityWorkshop (CCSW), 2009.

Author Profile

Jessy Cherian received the B.E. degree in Computer Science and Engineering from MahaBarathi Engineering College, Chinnasalem at 2012 and pursuing M.Tech in Computer Science and Engineering in Mount Zion College of Engineering, Kadammanitta.