International Journal of Scientific Engineering and Applied Science (IJSEAS) - Volume-1, Issue-3, June 2015
ISSN: 2395-3470
www.ijseas.com

ENABLING ANONYMOUS ENDORSEMENT IN CLOUDS WITH DECENTRALIZED ACCESS CONTROL

DEEPA.S[1] , Dr.M.SIVARAM[2]

PG SCHOLAR[1] , PROFESSOR[2]

[1,2]DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SELVAM COLLEGE OF TECHNOLOGY NAMAKKAL

[I] ABSTRACT

A decentralized access control scheme for data storage in clouds that supports anonymous authentication authentication. In this scheme, the cloud checks the validity of the series without knowing the user's identity before storing data. It also has the added feature of access control in which only valid users are able to decrypt the stored information. This prevents replay attacks and supports conception, variation, and reading data stored in the cloud. It also supports user revocation . This is an important property because a user, revoked of its attributes, might no longer be able to write to the cloud. Moreover, our authentication and access control scheme is decentralized and robust, unlike other access control schemes designed for clouds which are centralized. The communication, computation, and storage overheads are comparable to centralized approaches.

[II] INTRODUCTION

In cloud computing, users can outsource their computation and storage to servers (also called clouds) using Internet. This frees users from the hasis highsles of maintaining resources on-site. Clouds can provide several types of services like applications (e.g., Google Apps, Microsoft online), infra-structures (e.g., Amazon's EC2, Eucalyptus, Nimbus), and platforms to help developers write applications (e.g ., Amazon's S3, Windows Azure).

Much of the data stored in clouds is highly sensitive,for example ,medical records and social networks.security and privacy are,thus very important issues in cloud computing. User privacy is also required so that the cloud or other users do not know the identity of the user. The clo ud can ho ld the user acco untab le fo r the d ata it outsources, and likewise, the cloud is itself accountable for the services it provides. The validity of the user who stores the data is also verified. Apart from the technical solutions to ensure security and privacy, there is also a need for law enforcement.

Efficient search on encrypted data is also an important concern in clouds. The clouds should not know the query but should be able to return the records that satisfy the query. This is achieved by means of searchable encryption [3], [4]. The keywords are sent to the cloud encrypted, and the cloud returns the result without knowing the actual keyword for the search. The problem here is that the data records should have keywords associated with them to enable the search. The correct records are returned only when searched with the exact keywords.

Using homomorphic encryption, the cloud receives ciphertext of the data and performs computations on the ciphertext and returns the encoded value of the result. The user is able to decode the result, but the cloud does not kno w what data it has operated on. In such circumstances, it must be possible for the user to verify that the cloud returns correct results.

Considering the following situation: A law student, Alice, wants to send a series o f reports ab o ut so me malpractices by authorities of University *X* to all the professors of University *X*, research chairs of universities in the country, and students belonging to Law department in all universities in the province . She wants to remain anonymous while publishing all

evidence of malpractice. She stores the information in the cloud. Access control is important in such case, so that only authorized users can access the data. It is also important to verify that the information comes from a reliable source. The problems of access control, authentication, and privacy protection should be solved simultaneously. We address this problem in its entirety in this project .

Access control in clouds is gaining attention because it is important that only authorized users have access to valid service. There are broadly 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 contains the list of users who are authorized to access data. This is not feasible in clouds where there are many users. In RBAC ,users are classified based on their individual roles. Data can be accessed by users who have matching roles. The roles are defined by the system. For example, only faculty members and senior secretaries might have access to data but not the junior secretaries. ABAC is more extended in scope, in which users are given attributes, and the data has attached access policy. Only users with valid set of attributes, satisfying the access policy, can access the data. For instance, in the above example certain records might be accessible by faculty members with more than 10 years of research experience or by senior secretaries with more than 8 years experience.

[III] EXIXTING APPROACH

The Data are accessed in centralized way on the basis of single KDC,where KDC means Key Distribution Center(KDC) which is responsible for the distribution of keys and attributes to the users.A single Key distribution center does not support for authentication. A single failure of KDC can affect the maximum of data in cloud storage. Unfortunately, a single KDC is not only a single point of failure but difficult to maintain because of the large number of users that are supported in a cloud environment. The other drawback was that a user can create and store a file and other users can only read the file. Write access was not permitted to users other than the creator.

[IV] PROPOSED APPROACH

An area where access control is widely being used is health care. Clouds are being used to store sensitive information about patients to enable access to medical professionals, hospital staff, researchers, and policy makers. To Maintain the large number of datas in cloud, the decentralized access control approaches is proposed. It involves many KDC's for the distribution of secret keys and attributes of all users. It is important to control the access of data so that only authorized users can access the data. Using ABE, the records are encrypted under some access policy and stored i n t he clo ud . Users are give n sets o f attrib utes and corresponding keys. Only when the users have matching set of attributes, can they decrypt the information stored in the cloud.By the use of ABS the authenticity and the privacy can be achieved. This decentralized scheme also allows writing multiple times which was not possible in the existing approach.

4.1 ASSUMPTIONS

We make the following assumptions in our work:

1. The cloud is honest-but-curious, which means that the cloud administrators can be interested in view-ing user's content, but cannot modify it. This is a valid assumption that has been made in [12] and [13]. Honest-but-curious model of adversaries do not tamper with data so that they can keep the system functioning normally and remain undetected.

2. Users can have either read or write or both accesses to a file stored in the cloud.
3. All communications between users/clouds are se-cured by secure shell protocol, SSH.

4.2 CONTRIBUTIONS

The main contributions are:

1. Distributed access control of data stored in cloud so that only authorized users with valid attributes can access them.
2. Authentication of users who store and modify their data on the cloud.
3. The identity of the user is protected from the cloud during authentication.
4. The architecture is decentralized, meaning that there can be several KDCs for key management.

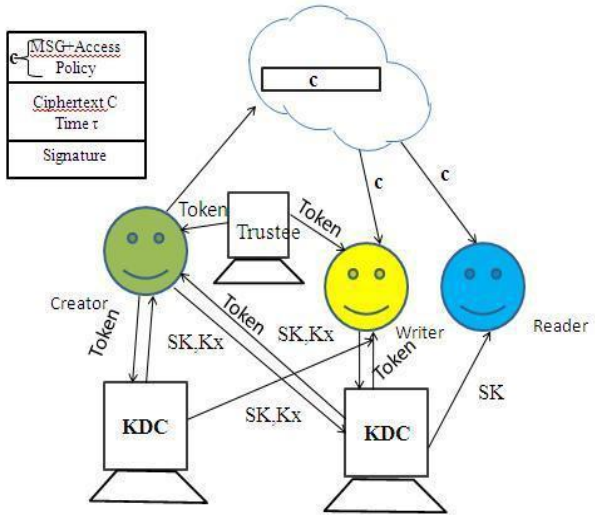
5. The access control and authentication are both collusion resistant, meaning that no two users can collude and access data or authenticate themselves, if they are individually not

International Journal of Scientific Engineering and Applied Science (IJSEAS) - Volume-1, Issue-3, June 2015
ISSN: 2395-3470
www.ijseas.com

authorized.

6. Revoked users cannot access data after they have been revoked.
7. The proposed scheme is resilient to replay attacks. A writer whose attributes and keys have been revoked cannot write back stale information.
8. The protocol supports multiple read and write on the data stored in the cloud.
9. The costs are comparable to the existing centralized approaches, and the expensive operations are mostly done by the cloud.

4.3 SYSTEM MODEL



Here is the privacy p reservin g authenticated access control scheme. According tothe scheme a user can create a file and store it securely in the cloud. This scheme consists of use of the two protocols ABE and ABS. There are three users, a creator, a reader, and writer. Creator 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 be scattered. 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, *S Ks* 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 ciphertext *C* with signature is *c*, and is sent to the cloud. The cloud verifies the signature and storesthe ciphertext *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.Write proceeds in the same way as file creation. By designating the verification process to the cloud, it relieves the individual users from time consuming verifications. When a reader wants to read some data stored in the cloud, it tries to decrypt it using the secret keys it receives from the KDCs. If it has enough attributes matching with the access policy, then it decrypts the information stored in the cloud.

4.4 ATTRIBUE BASED ENCRYPTION

Attribute-based encryption (ABE) is a relatively recent approach that reconsiders the concept of public-key cryptography. In traditional public-key cryptography, a message is encrypted for a specific receiver using the receiver's public-key. Identity-based cryptography and in particular identity-based encryption (IBE) changed the traditional understanding of public-key cryptography by allowing the public-key to be an arbitrary string, e.g., the email address of the receiver. ABE goes one step further and defines the identity not atomic but as a set of attributes, e.g., roles, and messages can be encrypted with respect to subsets of attributes (key-policy ABE - KP-ABE) or policies defined over a set of attributes (ciphertext-policy ABE - CP-ABE). The key issue is, that someone should only be able to decrypt a ciphertext if the person holds a key for "matching attributes" (more below) where user keys are always issued by some trusted party.

4.4.1 Ciphertext-Policy ABE

In ciphertext-policy attribute-based encryption (CP-ABE) a user's private-key is associated with a set of attributes and a ciphertext specifies an access policy over a defined universe of attributes within the system. A user will be able to decrypt a ciphertext, if and only if his attributes satisfy the policy of the respective ciphertext. arbitrary circuits).

CP-ABE thus allows to realize implicit authorization, i.e., authorization is included into the encrypted data and only people who satisfy the associated policy can decrypt data. Another nice features is, that users can obtain their private keys after data has been encrypted with respect to policies. So data can be

encrypted without knowledge of the actual set of users that will be able to decrypt, but only specifying the policy which allows to decrypt. 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 the data.

4.4.2 Key-Policy ABE

KP-ABE is the dual to CP-ABE in the sense that an access policy is encoded into the users secret key, e.g., (A∧C)∨D, and a ciphertext is computed with respect to a set of attributes, e.g., {A,B}. In this example the user would not be able to decrypt the ciphertext but would for instance be able to decrypt a ciphertext with respect to {A,C}.

An important property which has to be achieved by both, CP- and KP-ABE is called collusion resistance. This basically means that it should not be possible for distinct users to "pool" their secret keys such that they could together decrypt a ciphertext that neither of them could decrypt on their own

4.5 ATTRIBUTE BASED SIGNATURE

It is just not enough to store the contents securely in the cloud but it might also be necessary to ensure anonymity of the user. For example, a user would like to store some sensitive information but does not want to be recognized. The user might want to post a comment on an article, but does not want his/her identity to be disclosed. However, the user should be able to prove to the other users that he/ she is a valid user who stored the information without revealing the identity. There are cryptographic protocols like ring signatures [20], mesh signatures [21], group signatures [22], which can be used in these situations. Ring signature is not a feasible option for clouds where there are a large number of users. Group signatures assume the pre - existence of a group which might not be possible in clouds. Mesh signatures do not ensure if the message is from a single user or many users colluding together. For these reasons, a new protocol known as attribute-based signature (ABS) has been applied. In ABS, users have a claim predicate associated with a message. The claim predicate helps to identify the user as an authorized one, without revealing its identity. Other users or the cloud can verify the user and the validity of the message stored.

[V] CONCLUSION

It presented a decentralized access control technique with anonymous authentication, which provides user revocation and prevents replay attacks. The cloud does not know the identity of the user who stores information, but only verifies the user's credentials. Key distribution is done in a decentralized way. One limitation is that the cloud knows the access policy for each record stored in the cloud. In future, we would like to hide the attributes and access policy of a user.

[VI] REFERENCES

[1] S. Ruj, M. Stojmenovic, and A. Nayak, "Privacy Preserving Access Control with Authentication for Securing Data in Clouds," *Proc. IEEE/ACM Int'l Symp. Cluster, Cloud and Grid Computing*, pp. 556-563, 2012.

[2] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward Secure and Dependable Storage Services in Cloud Computing," *IEEE Trans. Services Computing*, vol. 5, no. 2, pp. 220-232, Apr.-June 2012.

[3] 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.

[4] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," *Proc. 14th Int'l Conf. Financial Cryptography and Data Security*, pp. 136-149, 2010.

[5] H. Li, Y. Dai, L. Tian, and H. Yang, "Identity-Based Authentica-tion for Cloud Computing," *Proc. First Int'l Conf. Cloud Computing (CloudCom)*, pp. 157-166, 2009.

[6] C. Gentry, "A Fully Homomorphic Encryption Scheme," PhD dissertation, Stanford Univ., http://www.crypto.stanford.edu/ craig, 2009.

[7] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-Based Cloud Computing," *Proc. Third Int'l Conf. Trust and Trustworthy Computing (TRUST)*, pp. 417-429, 2010.

[8] R.K.L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B.S. Lee, "Trustcloud: A Framework for Accountability and Trust in Cloud Computing," HP Technical Report HPL-2011-38, http://www.hpl.hp.com/techreports/ 2011/HPL-2011-38.html, 2013.

[9] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing," *Proc. Fifth*

ACM Symp. Information, Computer and Comm. Security (ASIACCS), pp. 282-292, 2010.

[10] D.F. Ferraiolo and D.R. Kuhn, "Role-Based Access Controls," *Proc. 15th Nat'l Computer Security Conf*, 1992.

[11] D.R. Kuhn, E.J. Coyne, and T.R. Weil, "Adding Attributes to Role-Based Access Control," *IEEE Computer*, vol. 43, no. 6, pp. 79-81, June 2010.

[12] M. Li, S. Yu, K. Ren, and W. Lou, "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 (SecureComm)*, pp. 89-106, 2010.

[13] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute Based Data Sharing with Attribute Revocation," *Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS)*, pp. 261-270, 2010.

[14] 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.

[15] 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.