

# Secure Data Stored in Clouds using Decentralized Access Control with Anonymous Authentication

Kh Tarkeshwari

M. Tech, CSE student  
T. John Institute of Technology  
Bangalore, India

Anju Abraham

Assistant Professor, Dept. of CSE  
T. John Institute of Technology  
Bangalore, India

**Abstract** – Research in cloud computing is receiving a lot of attention from both academic and industrial worlds. In cloud computing, users can outsource their computation and storage to servers (also called clouds) using Internet. This frees users from the hassles of maintaining resources on-site. Clouds can provide several types of services like applications (e.g., Google Apps, Microsoft online), infrastructures (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. The cloud is also prone to data modification and server colluding attacks. Access control in clouds is gaining attention because it is important that only authorized users have access to valid service. Here we propose a new decentralized access control scheme for secure data storage in clouds, that supports anonymous authentication. In the proposed scheme, the cloud verifies the authenticity of the server without knowing the user's identity before storing data. Our scheme also has the added feature of access control in which only valid users are able to decrypt the stored information. Only users with valid set of attributes, satisfying the access policy, can access the data. The scheme prevents replay attacks and supports creation, modification, and reading data stored in the cloud. We also address user revocation. 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.

**Keywords** – Access control, Authentication, Attribute-based signatures, Attribute-based encryption, Cloud storage.

## I. INTRODUCTION

Research in cloud computing is receiving a lot of attention from both academic and industrial worlds. In cloud computing, users can outsource their computation and storage to servers (also called clouds) using Internet. This frees users from the hassles of maintaining resources on-site. Clouds can provide several types of services like applications (e.g., Google Apps, Microsoft online), infrastructures (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. In one hand, the user should authenticate itself

before initiating any transaction, and on the other hand, it must be ensured that the cloud does not tamper with the data that is outsourced. User privacy is also required so that the cloud or other users do not know the identity of the user. The cloud can hold the user accountable for the data 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.

Recently, Wang *et al.* addressed secure and dependable cloud storage. Cloud servers prone to Byzantine failure, where a storage server can fail in arbitrary ways. The cloud is also prone to data modification and server colluding attacks. In server colluding attack, the adversary can compromise storage servers, so that it can modify data files as long as they are internally consistent. To provide secure data storage, the data needs to be encrypted. However, the data is often modified and this dynamic property needs to be taken into account while designing efficient secure storage techniques. 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. 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.

Security and privacy protection in clouds are being explored by many researchers. Wang *et al.* addressed storage security using Reed-Solomon erasure-correcting codes. Authentication of users using public key cryptographic techniques has been studied in.

Many homomorphic encryption techniques have been suggested to ensure that the cloud is not able to read the data while performing computations on them. Using homomorphic encryption, the cloud receives cipher text of the data and performs computations on the cipher text and returns the encoded value of the result. The user is able to decode the result, but the cloud does not know what data it has operated on. In such circumstances, it must be possible for the user to verify that the cloud returns correct results.

Accountability of clouds is a very challenging task and involves technical issues and law enforcement. Neither clouds nor users should deny any operations performed or

requested. It is important to have log of the transactions performed; however, it is an important concern to decide how much information to keep in the log. Accountability has been addressed in Trust Cloud. Secure provenance has been studied in.

Considering the following situation: A Law student, Alice, wants to send a series of reports about some 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 paper.

Access control in clouds is gaining attention because it is important that only authorized users have access to valid service. A huge amount of information is being stored in the cloud, and much of this is sensitive information. Care should be taken to ensure access control of this sensitive information which can often be related to health, important documents (as in Google Docs or Dropbox) or even personal information (as in social networking). 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 (ACL) 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. There has been some work on ABAC in clouds (for example, All these work use a cryptographic primitive known as Attribute Based Encryption (ABE). 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. 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 in the cloud. Users are given sets of attributes and corresponding keys. Only when the users have matching set of attributes, can they decrypt the information stored in the cloud. Access control in health care has been studied in. Access control is also gaining importance in online social

networking where users (members) store their personal information, pictures, videos and share them with selected groups of users or communities they belong to. Access control in online social networking has been studied in [19]. Such data are being stored in clouds. It is very important that only the authorized users are given access to those information. A similar situation arises when data is stored in clouds, for example in Dropbox, and shared with certain groups of people.

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, mesh signatures, group signatures, 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. ABS was proposed by Maji *et al.*. 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. ABS can be combined with ABE to achieve authenticated access control without disclosing the identity of the user to the cloud.

## II. RELATED WORK

ABE was proposed by Sahai and Waters. In ABE, a user has a set of attributes in addition to its unique ID. There are two classes of ABEs. In Key-policy ABE or KP-ABE (Goyal *et al.*, the sender has an access policy to encrypt data. A writer whose attributes and keys have been revoked cannot write back Stale information. The receiver receives attributes and secret keys from the attribute authority and is able to decrypt information if it has matching attributes. In Ciphertext-policy, CP-ABE, 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 allow only one KDC, which is a single point of failure. Chase proposed a multi-authority ABE, in which there are several KDC authorities (coordinated by a trusted authority) which distribute attributes and secret keys to users. Multi-authority ABE protocol, which required no trusted authority which requires every user to have attributes from at all the KDCs. Recently, Lewko and Waters proposed a fully decentralized ABE where users could have zero or more attributes from each authority and did not require a trusted server. In all these cases, decryption at user's end is computation intensive. So, this

technique might be inefficient when users access using their mobile devices. To get over this problem, Green *et al.* proposed to outsource the decryption task to a proxy server, so that the user can compete with minimum resources (for example, hand held devices). However, the presence of one proxy and one key distribution center makes it less robust than decentralized approaches. Both these approaches had no way to authenticate users, anonymously. Yang et al. presented, authenticate users, who want to remain anonymous while accessing the cloud.

To ensure anonymous user authentication Attribute Based Signatures were introduced by Maji et al. This was also a centralized approach. A recent scheme by the same authors takes a decentralized approach and provides authentication without disclosing the identity of the users. However, as mentioned earlier in the previous section it is prone to replay attack.

### III. THE ARCHITECTURE

The fig.1 shows the overall architecture and design . There are three users, a creator, a reader and writer. Creator receives a token  $\gamma$  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/his id (like health/social insurance number), the trustee gives her a token  $\gamma$ . 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. The message is encrypted under the access policy. The access policy decides who can access the data stored in the cloud.

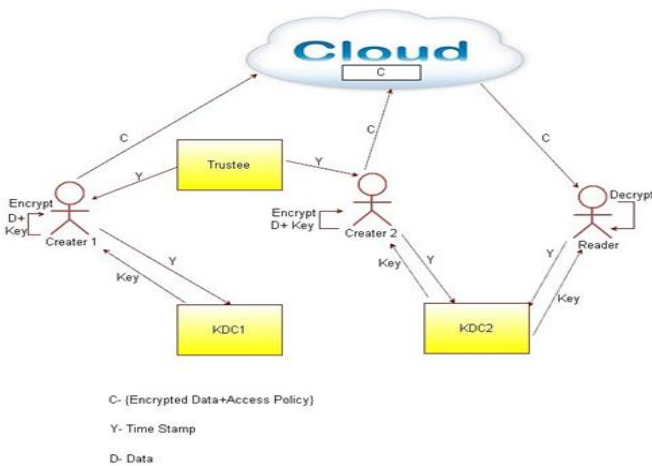


Fig 1. The architecture of SocioVision

The Ciphertext C with signature is c, and is sent to the cloud. The cloud verifies the signature and stores the 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.

### IV. CONCLUSIONS AND FUTURE WORK

We have 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.

It is an innovative idea where authorized users able to be compliant in secure way with access control mechanism. Also their identity is not revealed. Using Decentralized key distribution concept time delayed is reduced. This system presents a decentralized access control technique with anonymous authentication. 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.

### REFERENCES

- [1] C. Wang, Q. Wang, K. Ren, N. Cao and W. Lou, "Toward Secure and Dependable Storage Services in Cloud Computing", *IEEE T. Services Computing*, vol. 5, no. 2, pp. 220–232, 2012.
- [2] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy keyword search over encrypted data in cloud computing," in *IEEE INFOCOM*, pp. 441–445, 2010.
- [3] H. Li, Y. Dai, L. Tian, and H. Yang, "Identity-based authentication for cloud computing," in *CloudCom*, ser. Lecture Notes in Computer Science, vol. 5931. Springer, pp. 157–166, 2009.
- [4] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-based cloud computing," in *TRUST*, ser. Lecture Notes in Computer Science, vol. 6101. Springer, pp. 417–429, 2010.
- [5] 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. Available at <http://www.hpl.hp.com/techreports/2011/HPL-2011-38.html>.
- [6] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-based cloud computing," in *TRUST*, ser. Lecture Notes in Computer Science, vol. 6101. Springer, pp. 417–429, 2010.
- [7] S. Kamara and K. Lauter, "Cryptographic cloud storage," in *Financial Cryptography Workshops*, ser. Lecture Notes in Computer Science, vol. 6054. Springer, pp. 136–149, 2010.
- [8] S. Jahid, P. Mittal, and N. Borisov, "EASIER: Encryption-based access control in social networks with efficient revocation," in *ACM ASIACCS*, 2011.
- [9] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed access control in clouds," in *IEEE TrustCom*, 2011.
- [10] J. Hur and D. Kun Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems", *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 7, pp. 1214–1221, 2011.