Privacy Preservation in Cloud Using Attribute Based Encryption

T. Arokiaarun[1], **D. Saveetha**[2]

Department of Information Technology, SRM University, Chennai

Abstract: ***Security and privacy are very important issues in cloud computing. In existing system access control in clouds are centralized in nature. The scheme uses a symmetric key approach and does not support authentication. Symmetric key algorithm uses same key for both encryption and decryption. A single Key Distribution Centre (KDC) distributes secret keys and attributes to all users. A new decentralized access control scheme for secure data storage in clouds that supportsanonymous authentication is developed. The validity of the user who stores the data is also verified. The proposed scheme is resilient to replay attacks. In thisscheme. Data Owner can share the data and it's Key to the Permitted users. Data Sharing is achieved for three types of users. 1. User Based 2. Role Based (Position / Role), 3. Attribute (Experience). Data is uploaded by the Data Owner based on public key, secret key, Global key and Group key. Public key is randomly generated. Secret key & group key is generated via Attribute. Global key is generated randomly.We encrypt the uploaded file using ABE Algorithm.***

Keywords: Access control, Authentication, Key distribution Centre, ABE Algorithm.

1. Introduction

Cloud computing is service in which large groups of remote servers are networked to allow the centralized data storage, and online access to computer services or resources. Clouds can be classified as public, private or hybrid. 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 . Distributedaccess control of data stored in cloud so that only authorized users with valid attributes can access them. Authentication of users who store and modify their data on the cloud. The identity of the user is protected from the cloud during authentication. 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. 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 of information to keep in the log.

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.

Only users with valid set of attributes, satisfying the access policy. Attribute Based Signature (ABS) can be combined with ABE to achieve authenticated access control without disclosing the identity of the user to the cloud.

2. Objective

To propose a new privacy preserving authenticated access control scheme for securing data in clouds. In the existed scheme, the cloud data is not secure. A third person downloads the rfiles, they may see the record if they had the key which is utilized to decrypt the encrypted record. Once in a while this may be a failure because of the technology improvement and the programmers.

To overcome this issue there is a lot of procedures and techniques to make secure transaction and storage. In the proposed system, owner associates the set of attributes to the message by scrambling it with the comparing public key parts. Every client is assigned an access structure which is normally characterized as an access tree over information attributes .whoever satisfies the attributes andthe key, they only access the data from the cloud. Our scheme has added features of access control in which only valid users are able to decrypt the stored information.

3. 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 classifications of ABEs. In Key-policy ABE or KP-ABE, 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. All the approaches take a centralized approach and allow only one KDC, which is a single point of failure which required no trusted authority which requires every user to have attributes from at all the KDCs. The user who satisfies the attributes and key, they only access the data in cloud.

4. Existing Scenario

The security implementation details of the cloud in previous approach are described in a brief process below.

A. Identity Based Encryption

Identity-based systems allow any party to generate a public key from a known identity value such as an ASCII string. A trusted third party, called the Private Key Generator (PKG), generates the corresponding private keys. To operate, the PKG first publishes a master public key, and retains the corresponding master private key (referred to as master key). Given the master public key, any party can compute a public key corresponding to the identity ID by combining the master public key with the identity value. To obtain a corresponding private key, the party authorized to use the identity ID contacts the PKG, which uses the master private key to generate the private key for identity ID.As a result, parties may encrypt messages (or verify signatures) with no prior distribution of keys between individual participants. This is extremely useful in cases where pre-distribution of authenticated keys is inconvenient.

5. Attribute Based Encryption

It is a type of public-key encryption in which the secret key of a user and the cipher text are dependent upon attributes In such a system, the decryption of a ciphertext is possible only if the set of attributes of the user key matches the attributes of the ciphertext.A crucial security feature of Attribute-Based Encryption is collusion-resistance: An adversary that holds multiple keys should only be able to access data if at least one individual key grants access.

6. Proposed Solution

The main contributions of this paper are the following:

Distributed access control of data stored in cloud so that only authorized users with valid attributes can access them. The identity of the user is protected during authentication.

The architecture is decentralized, meaning that there can be several KDCs for key management. 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 authorized.

Revoked users cannot access data after they have been revoked. The proposed scheme is resilient to replay attacks. A writer whose attributes and keys have been revoked cannot write back stale information.

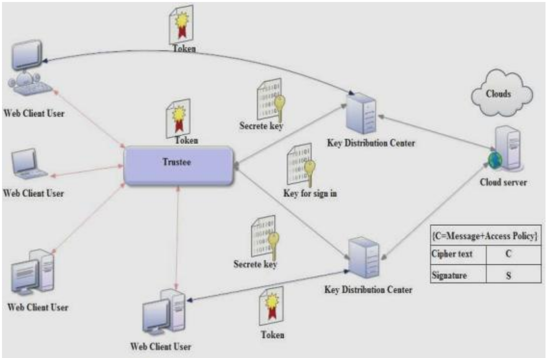

Figure 1: Secure Cloud storage model

The protocol supports multiple read and write on the data stored in the cloud. The costs are comparable to the existing centralized approaches, and the expensive operations are mostly done by the cloud.

1. Creation of KDC

Different number of KDC's are created for register a user details. User’s name, id and password are given as input to create KDC. Inputs will save in a database and to register a user details given a input as username and user id.

2. KDC Authentication

After KDC given a user id to a user, the user will enrolled the personal details to KDC's given a input as user name, user id, password etc. The

KDC will be verifying the user details and it will insert it in a Database.

3. Trustee and User Accessebility

Users can get the token from trustee for the file upload. After trustee is issuing a token, trustee can view the logs. User can login with their credentials and request the token from trustee for the file upload using the user id. After the user id received by the trustee, trustee will be create token using user id, key and user signature (SHA).

4. Creation Of Access Policy

After the key is received by the User, the message MSG is encrypted under the access policies. The access policies decide who can access the data stored in the cloud. 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 and user can upload the file after user get key from the KDC.

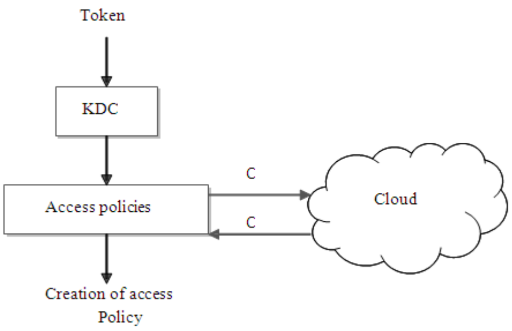

Figure 2: Creation Of Access Policy

5. File Accessing

Using their access policies the users can download their files by the help of KDC’s to issue the private keys for the particular users. After trustee token issuance for the users, the users produce the token to the KDC then the token verify by the KDC if it is valid then KDC will provide the public and Private key to the user. After users received the keys the files are encrypt with the public keys and set their Access policies (privileges).

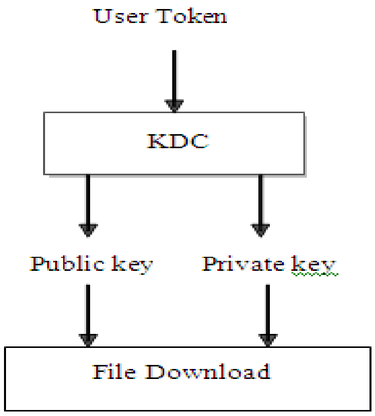

Figure 3: File Accessing

6. File Restoration

Files stored in cloud can be corrupted. So for this issue, using the file recovery technique to recover the corrupted file successfully and to hide the access policy and the user attributes.

7. Practical Setup

A. Data Storage in Cloud

The user also receivessecret keys s, k(x,u) for encrypting messages. The user then creates an access policy X which is a monotone Boolean function. The message is then encrypted under the access policy as

C = ABE.Encrypt(MSG,X). (1)

σ = ABS.Sign(Public key of trustee, Public key of KDCs,token, signing key, message, access claim)

The following information is then sent in the cloud.

c = (C, τ, σ, Y) (2)

B. Reading from the Cloud

When an user requests data from the cloud, the cloud sends the cipher text C using SSH protocol. Decryption proceeds using algorithm

ABE .Decrypt(C, {ski,u}) (3)

8. Security of the Protocol

we will prove the security of the protocol. We will show that our scheme authenticates an user who wants to write to the cloud. An user can only write provided the cloud is able to validate its access claim. An invalid user cannot receive attributes from a KDC, if it does not have the credentials from the trustee. If an user's credentials are revoked, then it cannot replace
data with previous stale data, thus preventing replay attacks.

9. Comparison with Other Techniques

As compared with ID based encryption method attribute based encryption is more secure. 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. Ourscheme also supports privacy preserving authentication, which is not supported by others.

10. Non-Functional Requirements

1. Perfomance

Performance testing is in accessing the data from the cloud. It's more secure than the previous methods. So, the system can perform well in the web environments.

2. Reliability

Reliability is one of the main advantage in this method. Every data storage and access the data from the cloud is reliable. Cloud both verified the authenticity and lifetime of the user who are all stored data in cloud. So it's more reliable.

3. Usability

Usability is the ease of use and learn ability of a software application. Usability is measure of how easy it is to use a product to perform prescribed tasks. Our software system also easy to use and user friendly .

4. Efficiency

The attribute based encryption can be performed each and every steps of verification. It most secure than the existing methods. So its efficiency is good. Our system gives good performance and resource behaviour .

11. Conclusion

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. So Its More secure than the previous method.

References

[1] S.Ruj, M.Stojmenovic and A. Nayak, "Privacy Preserving Access Control with Authentication for Securing Data in Clouds", *IEEE/ACM International Symposium on 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 T. Services Computing*, vol. 5, no. 2, pp. 220–232, 2012.

[3] 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.

[4] 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.

[5] 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.

[6] C. Gentry, “A fully homomorphic encryption scheme,” Ph.D. dissertation, Stanford University,2009.

[7] 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.

[8] R. K. L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B. S. Lee, “Trust cloud: 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.

[9] R. Lu, X. Lin, X. Liang, and X. Shen, “Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing,” in *ACM ASIACCS*, pp. 282–292, 2010.

[10] D. F. Ferraiolo and D. R. Kuhn, “Role-based access controls,” in *15th National Computer Security Conference*, 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, 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 multiowner settings,” in *SecureComm*, pp. 89–106, 2010.

[13] S. Yu, C. Wang, K. Ren, and W. Lou, “Attribute based data sharing with attribute revocation,” in *ACM ASIACCS*, pp. 261–270, 2010.