International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)
Volume 1, Issue 1, January 2014, PP 70-75
www.arcjournals.org

Anonymous Authentication for Secure Data Stored On Cloud with Decentralized Access Control

N Rajasekhar

Assistant Professor, Department of Computer Science and Engineering VNR VJIET, Hyderabad, India

Abstract: *We are introducing a new decentralized access control procedure which supports anonymous authentication for secure data storage in clouds. With-it the proposed scheme, the cloud verifies the series authenticity before storing data without knowing the user's identity. Our protocol also has the added access control feature in which only legitimate users can decrypt the stored information. Decentralized access control has several properties, such as privilege decentralized, policy definition decentralized, and allows application-defined methods and synchronization specifications, where the user may create another user under the user's subset and grant him unique access to the data by position or designation. Decentralized storage mechanism that is used to retrieve or receive data offers improved user authentication, device cancellation, such as revocation and replay attacks that can be triggered by delayed legitimate network data transmission. We developed a hospital management system where administrators, physicians, patients, labs, nurses are the consumers in the system where administrators have access to the data. For example: doctor can read data from the patient but can't change it so, admin just gives the doctor permission to read it. In this program we use KDC which KGC(key Generation Center) generates and groups. The system's main aim is to provide the data stored in the cloud with better protection and privacy, because the data stored in it is highly sensitive.*

1. INTRODUCTION

Decentralized storage device used to access or receive data offers improved user authentication, user cancellation, i.e. revocation and replay attacks triggered by delayed legitimate network data transmission. In access control we use KDC (key distribution center) to encrypt the data. Access control usually means that certain constraints are imposed to access the data. Which implies that only authorized users can access the data. Our system provides user verification in which only users who have been granted system can encrypt or decrypt the data. KGC plays a significant role in grouping the created KDC into the decentralized access control system. The data that is stored in the cloud can be updated or accessed through KDC access control. They created a hospital management system where physicians, patients, nurses, laboratory In-charge are the system users who will be given access to the data according to their positions.

In cloud computing, users can deploy their calculation or calculation and storage procedure to servers using the internet. Cloud has now replaced other systems for days and people are using it because it provides other services such as apps (Google Apps, Microsoft online), infrastruc-tures (Amazon's EC2, Eucalyptus, Nimbus), and tools to help developers create applications (Amazon's S3, Window

Azure) that make it exclusive to others. Total information or data stored in the cloud is highly sensitive and vulnerable, such as medical records, social networks, corporate ventures, etc., making protection and privacy a very significant cloud computingrequirement.

First, the person who uses it must verify or approve him herself before any action is taken and, second, it must be assured that the cloud does not cause any harm or change the data used. User privacy also plays a critical role in this program because other users or cloud will not be aware of the user's identicality. The cloud assumes user responsibility for the data used or stored, and is responsible for the services it delivers in the same way. These will also authenticate and check the identity of the user who modifies or stores data in the cloud. Cloud servers are more vulnerable to failures such as Byzantine failure where a part of the system can fail in the storage center and there may be incorrect information, resulting in failure that is hard to detect, data manipulation and center collusion attacks where multiple files or documents can collide and cause system harm. To give a better protected data repository, the data should be encrypted and later decrypted by authorized user for any modifications. Although, the data is frequently altered and due to this, dynamic property should also be considered during designing adequate secure storage methods. Adequate examination on encrypted data is also a vital point to be noted in cloud. The cloud should be designed in such a way that it should not know the desired question but it should be able to return the records accordingly to the question or desired query. This can be done only by searchableencryption.

The important words like keywords are transmitted to the cloud encrypted and cloud sends back the outcome without being aware of the actual keyword for the search. The issue over here is that the data files should have keywords connected with them to allow the search. Only when searched with the approximate keywords, the accurate files or data is sent back. To make sure that cloud is not allowed to read the data while executing computations on them, many homo-morphic encryption methods are proposed where the cloud gets ciphertext of the data and accomplishes calculations or computations on the ciphertext and sends back the encoded value of the outcome. The user will be allowed to decode the outcome, but the cloud does not know on which data it has worked on. The user must be able to validate that the cloud returns accurate outcome in such situations. Neither cloud nor users should reject or ignore any actions done or requested. So, it is vital to have a separate log of operationsachieved.

We will develop a system better than centralized system. There are many types of access control like User based access control, Role based access control, attribute access control. As user will have certain restrictions to access the data stored in centralized system, to overcome this problem We use KDC in decentralized system so that all the verified users can access the data from anywhere efficiently. It is a robust and secured system.

2. LITERATURE SURVEY

According to S. Kamara et al-Cryptographic Cloud Stor-age: The paper proposed improved protection for users, and storing their sensitive data in the storage of the cryptographic cloud. This allows for secure storage through encryption and decryption. Cryptography is essentially a method of securing information or data through codes and the information can only be read or written by the person who knows the code. While cryptography is highly secure it has a revocation problem that slows down cryptographic access control efficiency. They suggested a very new adequate revocation scheme to resolve this revocation which is highly efficient. The actual data will be first divided into a number of sections or slices in this process, and then sent to cloud storage. It helps to enhance privacy because the data owner has to retrieve just one slice when a revocation happens, then re-encrypt it instead of re-encrypting all the data that makes the process complicated. It can be sent back to cloud once again after re-encryption. The suggested revocation scheme can be applied for encryption based on the cipher of text policy attribute. Now cloud computing use for days has risen significantly. There are two types of cloud, namely private, public. The user’s data is highly protected in private cloud and all permissions are only provided by the user so that he does not need to worry about privacy. Privacy is the main problem in public cloud. So, cryptographic cloud storage has been implemented to handle this problem. This cryptographic cloud storage provides- Confidentiality, Integrity, Availability, Reliability, Efficient, Data sharing.

According to S. Yu et al-Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing: This paper has suggested several ways to create a better security and privacy. Typically they use highly sensitive and confidential cryptographic cloud storage to prevent abuse of the data. But the issue will still not be fixed completely. The paper suggested combining techniques such as attribute- based encryption, proxy re-encryption and lazy encryption for good security in order to solve this problem. The goal of this paper is to retain fine-grained access control of files that data owners store in cloud servers. This allows the owner of the data to give the users exclusive access. This also means cloud servers are not aware of the contents of the data. As mentioned above, encryption based attribute consists of data having attributes associated with a public key. Users can access data by a data attribute defined access structure. In order to decrypt the ciphertext, a secret key is created in such a way that a user can access data only if he can achieve data attribute access structure which is data attributes should be associated with the access structure. The ciphertext can be encrypted via proxy key in proxy re-encryption, without even knowing the plaintext. According to B. Waters et al- Ciphertext-Policy Attribute-Based Encryption: an Expressive, Efficient, and Provably Secure Realization: The paper proposed cloud computing actions with very low maintenance, efficient, and economical methods for cloud users to share data to community.

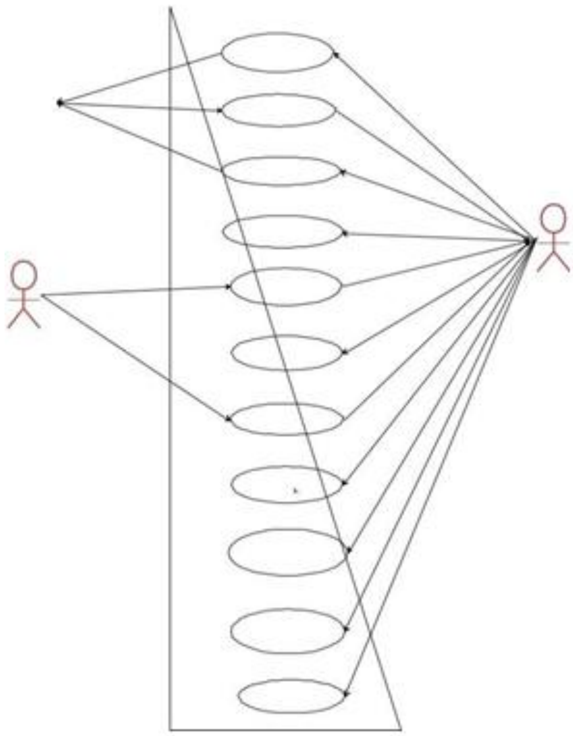
There is still a issue in the cloud about the multi-owner way of exchanging data when storing data and identity protection from a questionable domain. There could be simultaneous change in membership because of multi-owner. This paper focuses on safe and protected multi-owner data sharing known as Mona, which can be used in cloud-based dynamic groups with signature and dynamic broadcast encryption techniques to allow any user to exchange data with anyone anonymously. Under this method, the expense of measuring encryption does not depend on the number of users who have been canceled. They even take resolute proofs to examine system privacy. The paper's primary goal is to protect the data that will be processed from conspiracy attacks. The private key of the user is paired with data attributes and the cipher text can only be decrypted if the attribute fulfills the corresponding access matrix. They used linear secret sharing scheme. It aims at designing a complex access control with less number of parameters.

According to E. Goh et al – Sirius: Securing Remote Untrusted Storage: This paper introduced a stable file system in insecure network and P2P file systems such as NFS to be layered in. Managing any key and cancelation is simple with minimal communication Typically assuming that network storage can't be trusted and enforcing its own file sharing read-write access controls. For better file system SiRiUS uses hash tree constructions. Block server use is not needed because the SiRiUS uses a random access cryptographic file system. It uses large community NNL key revocation which is SiRiUS extension. File sharing can only be accomplished through cryptographic access which uses its own read-write operations. The design of the device is achieved by adding protection to the already used network file or new network file systems without even altering the software or hardware structure. Main Security Characters in SiR-iUS are: Confidentiality, Integrity, Untrusted file server, File access controls, Keymanagement, Keydistribution, Keyrevocation, Freshness guarantees.

According to B. Sheng et al –Verifiable Privacy-Preserving Range Query in Two-Tiered Sensor Networks: In this paper we define the structure of two tiered sensor networks. These sensor networks are the nodes of storage which function as tier between the sensors and use a sink to store data. Attackers may find interesting storage nodes which are used in structure. This paper proposed a SafeQ, a protocol that can be used to prevent attackers from accessing data on both sinks. The sink advantage is that it can also detect mal-functions performed by attackers or storage nodes, making SafeQ easy to store safely. Due to other purposes, the storage nodes are included in the framework- If fetched data is sent instantly by sensor nodes to the sink, too much energy may be needed and communication traffic in the sink canoccur.

Energy can be saved by sending the fetched data to the nearest storage node, rather than sending it directly. Since data is stored in storage nodes, the sensors can be limited to certain memory. Analysis of query will become simple if sink interacts directly with storage nodes rather than sensor nodes. Because storage nodes serve as an intermediate between sink nodes and sensor nodes, protection and privacy become a issue. The solution to this issue is to encrypt the data obtained from sensor nodes without even having knowledge of the specific values of the nodes being processed.

Trustee	Registration	
	Key generation	
	Login	User
Key distribution center	Share file	
	Encryption by sender	
	Decryption by receiver	
	Verify signature	
	Read/download file	
	Upload file	
	search	
	Logout	



3. Proposed Methodology

There are mainly four modules in this system which are as follows: KDC Module, Trustee Module, Signature Module

A. *KDC Module*

We emphasize that clouds should take a decentralized approach while distributing secret keys and attributes to users. It is also quite natural for clouds to have many KDCs in different locations in the world. The architecture is decentralized, meaning that there can be several KDCs for key management.

B. *Trustee Module*

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, which can be scattered. For example, these can be servers in different parts of the world.

C. *SignatureModule*

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 stores the ciphertext C. When a reader wants to read, the cloud sends

D. If the user has attributes matching withaccess policy, it can decrypt and get back originalmessage.

4. Implementation

A. *Systeminitialization*

Take a prime q, and G1 and G2 groups. The mapping is specified as e: G1/G1/G2. Let g1, g2 be G1 generators and hj be G2, j [tmax], arbitrary tmax generators. Let H have function as a hash. Let A0 = ha0 0, where randomly a0 ZZq is selected. (TSig, TVer) means Tsig is the private key for the signing of a document and TV er is the public key used for authentication. The trustee's hidden key is TSK = (a0, Tsig), and the public key is TPK = (G1,G2,H, g1,A0, h0, h1, .; htmax, g2, TVer).

B. Userregistration

The KDC draws at random KbaseG for a user with Uu identity. Let K0 = base K1/a0 The following token isoutput

= (u,Kbase,K0,), where is signature on u||Kbase using the signing key Tsig.

C. KDCsetup

As the designed system is decentralized, it can have several KDC's for keys at different locations.

D. Attribute Generation

Using the signature verification key TV er in TPK, the token verification algorithm verifies the signature found in:

This algorithm extracts Kbase from ASK[i] using (a, b) and calculates the base of Kx= K1/(a+bx), x xJ[i, u]. Key Kx can be checked for consistency using the ABS.KeyCheck(TPK, APK[i], ,Kx) algorithm, whichchecks e(Kx, AijBxij) = e(Kbase, hj), for all x J[i, u] andj[tmax

E. Sign

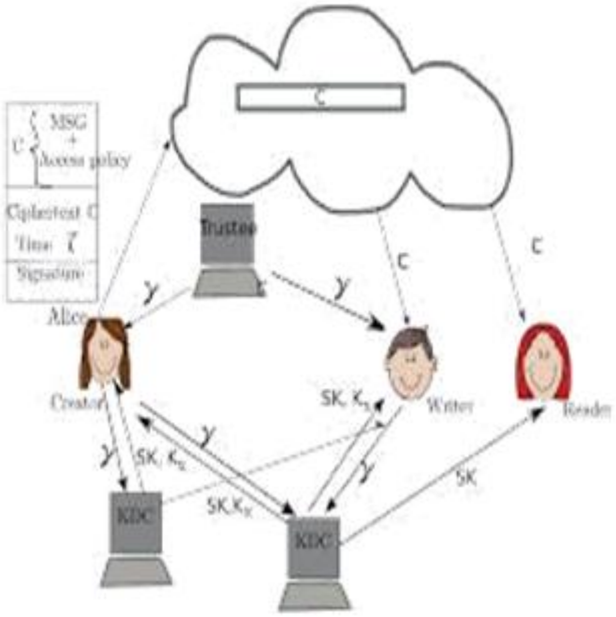
The access policy dictates who can access the cloud- saved data. The author wants to prove her validity on a claim policy Y, and signs the message under that claim. The signature ciphertext C is c, and is forwarded to the cloud. The cloud checks the signature, and stores the C ciphertext. When a reader decides to read, C is sent out by the cloud. Where the user has attributes that suit theaccess.

F. Verify

This relieves the individual users from the time-consuming testing process to the cloud. If a reader tries to read any data that is stored in the cloud, they attempt to decrypt it using the secret keys that they obtain from the KDC

5. CONCLUSIONS

Thedecentralizedaccess control with anonymous authentication for cloud storage is built for a secure data storage as compared with centralized systems. In this system the files have access controls to either read orwrite the file. We use KDC so that it generates secretkey



Every user in this system can maketheirdecisions and permission to access any file according to the user. As it supports anonymous authentication of data in cloud, only verified users can access the data without knowing the user's identity by the cloud or other users. Decrypting of data stored in cloud can be done only by authorized users. This scheme avoids replay attacks. In this scheme, every second a new key will be generated so that the data is highly secured.

REFERENCES

[1] S. Kamara and K. Lauter, “Cryptographic Cloud Storage,” Proc. 14th Int’l Conf. Financial Cryptography and Data Security, pp. 136- 149,2010.

[2] Chen, F., Liu, A.X.: SafeQ: Secure and Efficient Query Processing in Sensor Networks. In: Proceedings of the 29th IEEE International Conference on Computer Communications, INFOCOM 2010, pp. 1– 9. IEEE, California(2010)

[3] Bethencourt, J., Sahai, A., Waters, B.: Ciphertext- policy attribute based encryption. In: IEEE Symposium on Security and Privacy, pp. 321–334 (2007)

[4] H. Li, Y. Dai, L. Tian, and H. Yang, “Identity-Based Authentication for Cloud Computing,” Proc. First Int’l Conf. Cloud Computing(CloudCom),pp.157-166,2009.

[5] 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-June2012