IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Decentralized Access Control with Anonymous Authentication and Auto File Publishing in Cloud

Dr. A. N. Banubakode[1], Akshata Kulkarni[2], Harsha Kumble[2], Madhubala Chandak[2], Sanjana Prasad[2]

H.O.D, Information Technology, JSPM's Rajarshi Shahu College of Engineering, Pune, India[1]

Student, Information Technology, JSPM's Rajarshi Shahu College of Engineering, Pune, India[2]

Abstract: In today's world, the communication network is widely developed. You can send the texts as well as files, also it can be shared in one or many form. While communicating with the other person via medium, the registered details become transparent to the third party. What if we could demolish the transparency? We propose a cloud system where user can upload, download, view and also share data by keeping his identity anonymous. The application can be an anonymous sharing system for government where user can bring into notice the certain issues of government by keeping his identity secured. It is also important to ensure that respective file is delivered to the authorities in any case.

Keywords: anonymous sharing system, decentralized, access control, timer function, double authentication.

I. INTRODUCTION

Several trends are opening up and the era of cloud computing, which is an Internet-based development and uses of computer technology. The more powerful processors with the Software as a Service (SaaS) computing architecture are transforming the data centers into pools of computing services. The increasing network bandwidth and reliable yet flexible network connections make it possible that users can now subscribe to high quality services. Moving data into the cloud offers great convenience to the users as they don't have to care about the complexities of direct hardware management. The initial cloud computing vendors are Amazon Simple Storage Service (S3), and Amazon Elastic Compute Cloud (EC2). While these internet-based online services do provide huge amounts of storage space and adaptable computing resources, this platform shift how is eliminating the responsibility of local machines for data maintenance at the same time. In cloud computing a user can rent the storage and computing resources of a server (also known as cloud) which is provided by a company. Users only require a terminal, a smart phone or tablet which should be connected to the inter connected network. Instead of user's machine, the application runs on the cloud. Clouds can store large amount of data, so that mobile users do not have to carry their data. Some clouds provide application services like Google Apps, Microsoft online some provide infrastructural support like Amazon's EC2, Eucalyptus, Nimbus, or platform, to help developers to write applications that will run on the cloud such as Amazon's S3 or Windows Azure.

Security of data and privacy of users is more important and has to be preserved. Cloud should make sure that the users trying to access data and services are authorized ones. Authentication of users is achieved by using many public key cryptographic techniques. Users should also ensure that the cloud is not tampering with their data and computational results. Sometimes, it can also be important to hide the users identity for privacy reasons. For example, while storing some medical records, the cloud should not be able to access records of a particular patient, given the identity. Users should also ensure that the cloud is able to perform some computations on the data, but without knowing the actual data. Homomorphic encryption techniques are a way to hide data from clouds and also carry on computation on the data.

Most of the data stored in clouds is highly sensitive, such as medical records and social networks. Thus, security and privacy are very important issues in cloud computing. In one hand, the user authentication should be done before initiating any transaction, and on the other hand, the user should also make sure that the cloud does not tamper with the data that is outsourced. User privacy is also required so that the identity of the user is not known to the cloud or other users. The cloud has the capability to hold the user accountable for the data it outsources, and similarly, the cloud is itself accountable for the services that it provides.

The validity of the user who stores the data on the cloud is also verified. Apart from the technical solutions, to ensure security and privacy, there is a need for law enforcement. Efficient search on encrypted data is also very 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 searchable encryption. The keywords are sent to the cloud encrypted, and the cloud then returns the result without knowing the actual keyword for the search. The main problem here is that the data records should have some keywords associated with them to enable the search. The correct records are returned only if they are searched with the exact words.

DOI 10.17148/IJARCCE.2017.6222

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Many analysts are exploring security and privacy protection in clouds. Using homomorphic encryption, the cloud receives ciphertext of the data and then computations on the ciphertext are performed and then encoded value of the result is returned. The user is able to decode the result, but the cloud does not know what data it has operated on. In such situations, it must be possible for the user to verify that the cloud returns correct results.

Access control in clouds is gaining more attention because it is important that only authorized users should have access to the valid service. A huge amount of information is being stored in the cloud, and much of this information is sensitive information. Care should be taken to ensure that access control of this personal information which can often be related to health, important documents such as in Google Docs or Dropbox or even personal information (as in social networking). Access control is mainly classified into: user-based access control (UBAC), role-based access control (RBAC), and attribute-based access control (ABAC). In user-based access control (UBAC), the access control list contains the list of users who are authorized to access the data. This is not feasible in clouds where there are more than one users. In role-based access control (RBAC), users are classified based on their individual roles. Data can be accessed by only those users who have matching roles. The roles have been interpreted by the system. For example, only faculty members and senior persons might have access to data but not the junior persons. Attribute-based access control (ABAC) is more extended in scope, in which users are given attributes, and the data has attached access policy.

Users which have valid set of attributes and which satisfy the access policy, only those users can access the data. For instance, in the above example, certain records can be accessible by faculty members with more than 10 years of research experience or it can be accessed by senior secretaries with more than 8 years experience. The pros and cons of RBAC and ABAC are as discussed below. Some work is done on ABAC in clouds. All these work use a cryptographic primitive known as attribute based encryption (ABE) technique. The extensible access control markuplanguage has been suggested for ABAC in clouds.

Only storing the contents securely in the cloud is not enough, but it might also be necessary to ensure the anonymity of the user. For example, a user would like to store some sensitive information but the user also does not want to get recognized. The user might want to post a comment on an article, but the user does not want to disclose his/her identity. However, the user has the responsibility to prove to the other users that he/ she is a valid user who stored the information without revealing the identity.

An effective and flexible distributed storage verification strategy with explicit dynamic data support to ensure the correctness and availability of users' data in the cloud is proposed. The focus is on erasure correcting code in the file distribution system to provide guarantee of data dependability against Byzantine servers where a storage server may fail in arbitrary ways is done. Traditional replication-based file distribution techniques reduces the communication and also storage overhead. By using the similar token with distributed verification of erasure-coded data, our strategy is to achieve the storage correctness insurance as well as data and storage error localization: whenever data corruption is detected during the storage correctness verification, our scheme also can guarantee the identification of the misbehaving server(s). A good balance between error resilience and data dynamics should be maintained. For this, we further explore and demonstrate how to support dynamic operation on data blocks efficiently, while maintaining the same level of storage correctness assurance. It can be summarized as the following three aspects: 1) Compared to many of its previous scheme, which only provide some conclusion regarding the status of storage across the distributed servers, this has been overcome in the proposed scheme that achieves the integration of storage correctness insurance and data error localization, from which the misbehaving of server(s) can be identified. 2) Unlike most prior works for ensuring remote data integrity, secure and efficient operations including delete, update and append is supported by the new scheme. 3) The experiment results demonstrate the proposed scheme is highly efficient.

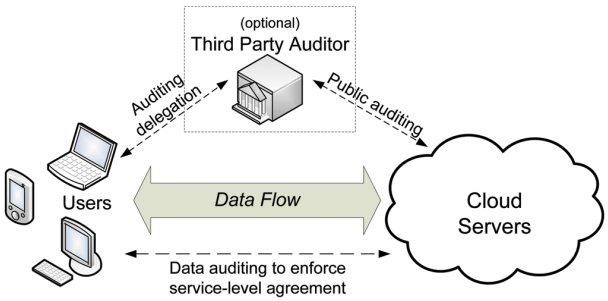


Fig 1-Cloud storage service architecture

II. OUR CONTRIBUTIONS

The main contributions of this paper are as given below:

1.There is Distributed access control of data stored in cloud so that only authorized users with valid attributes can access them.

2.It is also important to ensure that respective file is delivered to the authorities in any case. This is made possible by the addition of a timer function which will automatically upload the file after a specific amount of time

3. The identity of the user is protected from the cloud during authentication.

4. Revoked users cannot access data after they have been revoked.

5. The proposed scheme is resilient to replay attacks. A writer whose attributes and keys have been erased cannot write back stale information.

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

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

7. The costs are comparable to the existing centralized approaches, and the expensive operations are mostly done by the cloud.

III. RELATED WORK

Sahai and Waters proposed ABE [26]. In addition to its unique ID, ABE also has a set of attributes. There are two classes of ABEs. In key-policy ABE or KP-ABE (Goyal et al. [27]), a policy to encrypt data is provided to the sender. Stale information cannot be written back by a user (writer) whose attributes and keys have been revoked. The receiver is able to decrypt information if he/she acquires matching information. This is possible because the attribute authority can provide the receiver with attributes and secret keys. In Ciphertext-policy, CP-ABE ([28], [29]), the receiver has the access policy in tree form 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 [30] proposed a multiauthority ABE, in which there are several KDC authorities (coordinated by a trusted authority) which is used for distributing attributes and secret keys to users. A study on Multiauthority ABE protocol [31] and [32], which required no trusted authority which requires every user to have attributes from at all the KDCs. Recently, a fully decentralized ABE, where users could have zero or more attributes from each authority and did not require a trusted server was proposed by Lewko and Waters[35]. In all these cases, decryption is computation intensive at user's end. So, there might be some inefficiency in this technique when users access using their mobile devices. To get over this problem, Green et al. [33] proposed of outsourcing the decryption task to a proxy server, so that computation can be done with minimum resources by the users. However, the presence of one KDC and one proxy and one KDC reduced the robustness when compared to the robustness provided by decentralized approaches. Both these approaches had no way to authenticate users, anonymously. Yang et al. [34] presented a modification of [33], to authenticate users, who, while accessing the cloud, wanted to remain anonymous. To ensure anonymous user authentication ABSs were introduced by Maji et al. [23]. This was also a centralized approach. A recent scheme by Maji et al. [24] takes a decentralized approach and provides authentication without disclosing the identity of the users. However, replay attacks are still possible as mentioned earlier in the previous section. This will help us to better appreciate why Lewko and Water's scheme [18] is best suited for data access control in clouds. ABE was proposed by Sahai and Waters [19]. In ABE, a user is provided with a set of attributes along with unique ID. Identity-based encryption (IBE)which was proposed by Shamir [20] has been extensively studied. Each user in an IBE scheme has a unique identity, and the unique information about the user is the public key. IBE can be termed as a special case of ABE. There are two classes of ABE. In Key-policy ABE or KPABE (Goyalet al[21]), the sender is provided with an access policy to encrypt data. The receiver gets attributes and secret keys from the attribute authority and decryption of information is possible if it has matching attributes. In Ciphertext-policy, CP-ABE (Bethencourt et al. [22]), the access policy in the form of tree, with attributes as leaves and monotonic access structure with AND, OR and other threshold gates lies with the user. Although, it is an assumption that the attribute authority (KDC) in all mentioned protocols is honest. However, that might not be the case, because in a distributed system, authorities can fail or be corrupt. A multi-authority ABE was proposed by Chse [23], in which several KDC authorities are present (coordinated by a trusted authority) which distribute attributes and secret keys to users. Chase and Chow [24] devised a multi-authority ABE protocol in which there was no requirement of a trusted authority. However, a major problem was that a user required the authorities to grant at least one attribute, which might not be practically possible. Recently, Lewko and Waters [18] proposed a fully decentralized ABE, where users could have none or more attributes from each authority and there was no requirement of a trusted server. Their protocol has been recently applied to gain access control in intelligent transport system [25]. This enables vehicles to transmit messages, in such a way that only authorized vehicles can receive them.

IV. WORKING

Cloud is gaining more popularity because only authorized users are allowed to access the cloud .Cloud allows us to store sensitive information that to be in a huge amount. The information stored in in clouds can be related to health, important documents or even personal information therefore proper care must be taken to ensure access to this information.

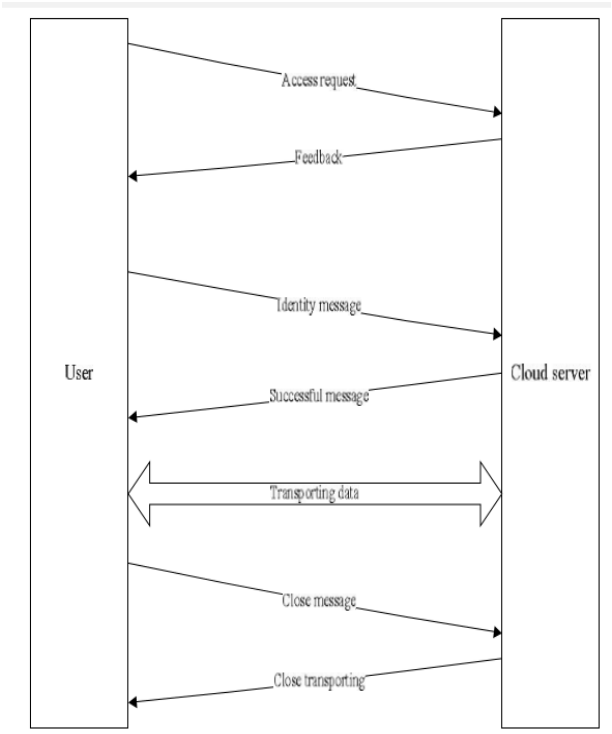
The user has to provide username and password then only the client get itself authenticated. For encrypting the file, a private key is generated using various combinations of username and password. The client request to key manager for the public KEY.

The work of key manager is to verify the file using the policy associated with the file .Public key will be generated only if the policy matches with the file name otherwise new public key will be generated. The file will be encrypted and uploaded into the cloud with the public key and private key.

To download the file the user is first authenticated. If the authentication stage is success, then the file will be downloaded to the user. Still the user is unable to read the contents of file. The user should request for the public key from key manager. After authentication, the key manager will provide the public key to the user.

In any case the file should be delivered to the right authorities. A predefined deadline is scheduled after which the file will be automatically posted even without the provider's permission.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017



Authentication and login part:
1) Modular Structure: The system has different modules having different functions.
2) User Registration: To be able to use the cloud resources each user need to provide his/her information to the authentication database in order to register.
3) User Login: In this is stage the user provides the authentication credentials. The user is identified by matching the credentials with the registration database. This will help in identifying whether the user involved in data theft or unauthorized access.
4) Access Control: Owner in this system can only allow or deny access to their data for some other user in the system. The user has access to the data only if the owner permits. User may further have the facility to limit this access for a particular time .
5) Encryption & Decryption: Before uploading files to the cloud, the files are encrypted. Similarly while accessing the contents of files, it needs to be decrypted. This technique of encryption and decryption adds security to a greater extent. The owner can have the right to encrypt data segments with different keys and then provide the users the keys only to which he wants to restrict the access to the user.
6) File Upload: In this stage the encrypted file is uploaded by the owner.This file is saved in a remote database and can be downloaded by a registered and authorized user only. During download, the data can be decrypted only with the key it was encrypted with.
7) File Download: Once authorized the user can download and decrypt the required data. The decryption key needs to be provided by the owner.
8) Cloud Service Provider Registration: Like user and owner, cloud manager also has to register in a similar way. Cloud service provider or manager is the person that can access all the data on cloud. This entity is like a trustee and needs to be trusted by user and owner.

V. COMPARISON WITH OTHER ACCESS CONTROL SCHEMES IN CLOUD.

We compare our scheme with other access control schemes and show that our scheme supports many features that the other schemes did not support. 1-W-M-R means that only one user is allowed write while many users can read. M-W-M-R means that many users can write and many users can read. As most schemes do not support many write operation which is supported by our scheme. Most of the others schemes are centralized but our scheme is robust and decentralized. Our scheme also supports privacy preserving authentication and auto file publishing in cloud which is not supported by others.

Most of the schemes do not support user revocation, which is provided by our scheme. We compare the computation cost and communication costs incurred by the users and clouds and show that our distributed approach has comparable and cost efficient than other centralized approaches. The most expensive operations involving pairings and this process are done by the cloud. If we compare the computation load of user of different schemes during read we see that our scheme has comparable costs.

VI. CONCLUSION

We have presented a decentralized access control technique with anonymous authentication and auto file publishing, which provides user revocation user anonymity and prevents replay attacks. The cloud does not know the identity of the user who stores information, but only verifies the user's credentials therefore providing user anonymity. Key distribution by key manager is done in a decentralized way. One limitation is that the cloud service provider knows the access policy for each file stored in the cloud. In future, we would like to implement a scheme that is able to hide the attributes and access policy of a user.

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. 556563, 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. 136149, 2010.
[5] 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.

[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,
[12] June 2010.
[13] 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.
[14] 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.
[15] 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.
[16] 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.
[17] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed Access Control in Clouds," Proc. IEEE 10th Int'l Conf. Trust, Security and Privacy in Computing and Communications (TrustCom), 2011.
[18] S. Jahid, P. Mittal, and N. Borisov, "EASiER: Encryption-Based Access Control in Social Networks with Efficient Revocation," Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS), 2011.
[19] R.L. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret," Proc. Seventh Int'l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 552-565, 2001.
[20] X. Boyen, "Mesh Signatures," Proc. 26th Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 210-227, 2007.
[21] D. Chaum and E.V. Heyst, "Group Signatures," Proc. Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 257-265, 1991.
[22] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-Based Signatures: Achieving Attribute-Privacy and Collusion-Resistance," IACR Cryptology ePrint Archive, 2008.
[23] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-Based Signatures," Topics in Cryptology - CT-RSA, vol. 6558, pp. 376-392, 2011.
[24] Beimel, "Secure Schemes for Secret Sharing and Key Distribution," PhD thesis, Technion, Haifa, 1996.
[25] Sahai and B. Waters, "Fuzzy Identity-Based Encryption," Proc. Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 457-473, 2005.
[26] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conf. Computer and Comm. Security, pp. 89-98, 2006.
[27] Sahai and B. Waters, "Fuzzy Identity-Based Encryption," Proc. Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 457-473, 2005.
[28] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conf. Computer and Comm. Security, pp. 89-98, 2006.
[29] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption," Proc. IEEE Symp.Security and Privacy, pp. 321-334, 2007.
[30] X. Liang, Z. Cao, H. Lin, and D. Xing, "Provably Secure and Efficient Bounded Ciphertext Policy Attribute Based Encryption,". ACM Symp.Information, Computer and Comm. Security (ASIACCS), pp 343-352, 2009.
[31] M. Chase, "Multi-Authority Attribute Based Encryption," Proc. Fourth Conf. Theory of Cryptography (TCC), pp. 515-534, 2007.
[32] H. Lin, Z. Cao, X. Liang, and J. Shao, "Secure Threshold Multi-Authority Attribute Based Encryption without a Central Authority," Proc. Progress in Cryptology Conf. (INDOCRYPT), pp. 426-436, 2008.
[33] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-Based Signatures," Topics in Cryptology - CT-RSA, vol. 6558, pp. 376-392, 2011.
[34] Beimel, "Secure Schemes for Secret Sharing and Key Distribution," PhD thesis, Technion, Haifa, 1996.