IJESMR
International Journal OF Engineering Sciences & Management Research

DISTRIBUTED ATTRIBUTE BASED ACCESS CONTROL FOR CLOUD STORAGE

Swati S. Gore*1, Gajanan S. Deokate2, ShyamraoV.Gumaste3
[*1] ME Student Department of computer Engineering, SPCOE, Dumbarwadi, otur, India
[2]Assistant Professor Department of computer Engineering, SPCOE, Dumbarwadi, otur, India
[3]Assistant Professor Department of computer Engineering, SPCOE, Dumbarwadi, otur, India

Keywords: Access Control, Homomorphic Encryption, SHA-1, Cloud Storage.

ABSTRACT

We propose a secure cloud storage scheme that addresses security, privacy, authentication issues in the cloud. We are providing anonymous authentication feature so that user who want to store data on cloud remains anonymous from the cloud and other user. It is attribute based access control scheme so user with valid set of attributes can access the data on cloud. For key generation and management we are using distributed approach. The key distribution centers are multiple which manages keys so it is distributed in nature. We are using homomorphic encryption scheme which uses paillier encryption algorithm which is asymmetric algorithm and for authentication we are using SHA-1.

INTRODUCTION

Cloud computing is a new computing area where computer processing is performed through internet by different browsers such as Firefox, internet explorer etc. Cloud computing builds on established trends for driving the cost out of the delivery types of services with increasing the speed and duration with which services are deployed. It reduces the time from initiation of application architecture to actual deployment. As cloud computing has become important, more and more sensitive data is being centrally stored into the cloud by users. To protect the sensitive data from attacker, the data should be in encrypted form before uploading on the cloud. However, this gives a new problem for performing search operation over the encrypted data efficiently. Although the most of searchable encryption techniques allow a user to search on the encrypted data by providing confidentiality, these solutions are not useful for the verification process of searched result. A cloud server may be selfish in order to save its computation ability and its bandwidth. For example, it may execute only a part of the search and returns only some part of the searching result.

Today’s computing techniques have attracted more people to store their important data on third-party servers either for sharing easiness or for reducing cost. When people uses features of these new emerging technologies and services invented, their concerns about data security also important. Usually, users would like to make their important and confidential data only accessible to some authorized users. In many cases, it is also important to provide various access services such that data access policies are defined using user attributes and roles. We can easily find that these security points and requirements would become more important in the coming days of cloud computing wherein organizations, individuals, and businesses may put their various types of data, including the highly sensitive data, into the cloud. Existing access control strategies will not be as effective under this new paradigm because the service providers and the data owners may possibly belong to different trusted domains, and the third-party storage servers themselves may not be fully trusted.

To obtain trust on cloud, security measurements in the cloud must be provided to the users. There is always a situation that the cloud environment is secured with respect to some requirements and the users are searching for a various set of security. The important thing is to see that the cloud provider meets the security requirements of the application. In order to have a secured cloud computing, we have to consider the different areas like architecture of cloud computing, interoperability, portability, security, business continuity, data center operations, Application Security, Key management and encryption, identity and access management. The reason why users are very afraid from the safety of self-data which is saved in the cloud is that they don’t know who is managing it while in the server of the cloud computing service provider. Most of the users using the cloud computing applications like storing their data files on the server to access it anywhere when they required through internet, do not care much about the security of their data files, those documents are common files that don’t need to be secured. But in the case of big industries which have very important information to take care, they need to have secured cloud computing environment.

Access control techniques [2] are very important strategies to provide security for the system where only authorized user should able to access the resources they required. This is very important in cloud because high priority and sensitive data is being stored in cloud such as online document, personal information (Facebook, twitter), and medical information. Access control is a key point, because insider attacks is having a high risk. Any cloud user needs to know who is managing their data and what types of controls are applied to these individuals. The model of application centric access control, where most of the applications keeps list of its users and manages them, is not more feasible in cloud based architectures. This is because the user space maybe shared across various applications that can cause data replication. Also, it requires the user to remember multiple accounts/passwords and maintain them. Cloud needs a user centric access control approach where every user request to any service provider is provided with the user identity and related information. User identity will have identifiers or attributes that identity and define the user. The identity is along with domain, but is portable. User centric [2] approach leaves the user with the much control of their digital identifications. User centric approach also required that the system maintains a context of information for every user, so to find how best to react to in a given situation to a given user request. There are different types of access control methods. First is User based access control, a method in which there is a list of users who can access the data. Here there is need to maintain a list of large no of users so it is not feasible as number of users in cloud are more. Second is Role base access control method where the users with specific role can only access the data, means suppose in college Head of department and secretary can access the data not staff members. Third method is attribute based access control method. In the same above example by considering attribute as an experience staff with more than 5 year of experience can access the data.

Consider example, suppose a law student in university p want to show some reports about some malpractices by authorities of university p to other law student of university p, all the professors of university p research chairs of other universities in the country. She/ He wants to remain unknown form all this members while showing these reports of malpractices. Access control is important here as only authorized users can have access the data. It is also important point that information shown comes from reliable source. So here in this paper we are solving the problems of access control, authentication and privacy preservation.

Existing work [9],[10] on access control in cloud infrastructure is centralized in nature as they have used single key generation center. They use ABE, attribute based encryption scheme for more security. Using ABE protocol the data is encrypted under some access policy and stored in the cloud. Users are given different set of attributes and required keys, users with sufficient matching attribute can decrypt this data stored in the cloud. In existing system authors use centralized approach where single key distribution Centre (KDC) distributes secrete keys and attributes to all users. In this case a single KDC is not only a single point of failure but also it is difficult to maintain because there are large no of users in the cloud environment. So here in this system decentralized approach is used while generating secrete keys. In existing work Ruj et al [1] proposed a distributed access control scheme in clouds. However the scheme did not give user authentication. Another limitation was that a user can create and store a file and other users can only able to read the file. Write access was not permitted to users other than the creator. In our scheme revocation of users is also addressed so that revoked user cannot access the data after removed. So this scheme prevents from different replay attacks. A writer whose attributes and keys are revoked from system cannot able to write back stale information.

For gaining authentication of users many cryptographic techniques are used. Using signatures authentication is done and it also helps in detection of unauthorized modifications of data. Digital signature scheme provides framework for providing digital signatures. Also group signature, mesh signature, ring signatures [1] can be used in this situations. As there are large number of users in cloud environment these are not feasible options.

LITERATURE SURVEY

Sahai and water [4] proposed Identity-Based Encryption (IBE) scheme which is known as Fuzzy Identity-Based Encryption. In Fuzzy IBE user can view an identity as set of descriptive attributes. A Fuzzy IBE scheme make use of a private key for an identity I, to decrypt a cipher text encrypted with an identity I', if and only if the identities I and I' are close to each other as measured by the set overlap metric. A Fuzzy IBE [4] scheme can be applied to enable encryption using biometrics inputs as identities of user. The error tolerant feature of a Fuzzy IBE scheme is precisely what allows for the use of biometric identities, which may have some noise each time they are sampled. Moreover Fuzzy-IBE can be used for a type of application that can be termed as attribute-based encryption [ABE]. ABE has two types, first as a KP-ABE and second as a CP-ABE.

In KP-ABE cryptosystem [5], cipher texts are labeled with sets of attributes and private keys are associated with access structures that control which cipher texts a user is able to decrypt. Since the access structure is specified in the private key, while the cipher texts are associated with a set of descriptive attributes. One can build a tree access structure where the interior nodes consist of AND, OR gates and the leaves consist of different parties. Any party that satisfies the tree can reconstruct the secret. For instance, if any user has the key associated with the access structure L AND M, and another user has the key associated with the access structure M AND N, we would not want them to be able to decrypt a cipher text whose only attribute is M by colluding.

CP-ABE [6] maintains a list of users U and a list of attributes A that defines those users. All users will be given a set of attributes that decides their privileges. When a manager wants to encrypt a file, he will first construct the access policy as a tree where the leaves are the attributes that allows the users to access the file. The root will contain the secret key that can decrypt the file. So when a user tries to access a file, the system will match his attributes that associated with his key. If those attributes satisfies the access policy associated with the file, the system will decrypt the file, otherwise it will not be decrypted.

All these schemes are centralized in nature that allows a single KDC to generate keys. So chase proposed a multiauthority ABE scheme [7]. To overcome the drawbacks of a single authority attribute-based system. This system uses a central authority (CA) and multiple attribute authorities (AAs). The problem with the Chase multiauthority attribute-based encryption system is that the CA can decrypt every cipher text which reduces the user privacy and confidentiality of user data. So bozovic et al [8] presented a multi-authority attribute based encryption scheme in which only the set of recipients defined by the encrypting party can decrypt a corresponding cipher text. The central authority is considered as honest-but-curious nature, on the one hand it honestly follows the protocol, and on the other hand it tries to decrypt arbitrary cipher texts. This scheme, which like its predecessors relies on the Bilinear Diffie-Hellman assumption, has a complexity comparative to that of Chase's scheme. Scheme is secure in the selective ID model and can tolerate an honest-but-curious central authority [8].

SYSTEM ARCHITECTURE

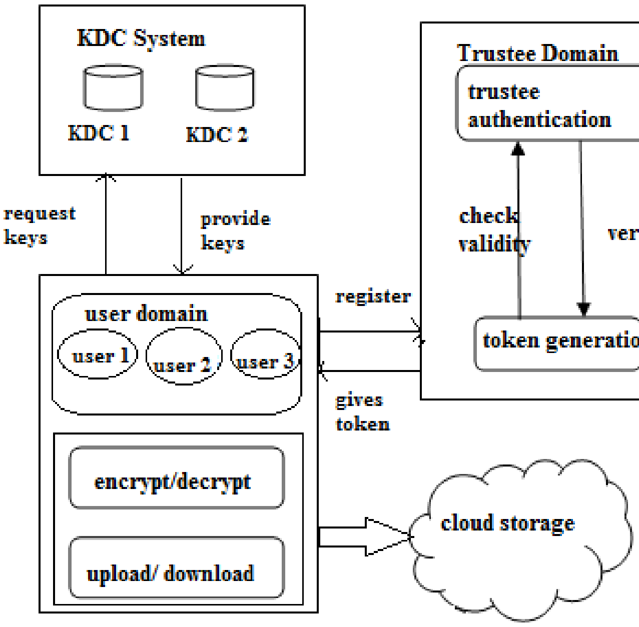

Fig 1. System Architecture

This section shows our privacy preserving distributed access control scheme. There are four main modules in this system, Trustee, KDC, Cloud Storage and client as a multiple users. Trustee is any third party authenticator like some government organization who assigns unique identifier to users. When users do registration, the trustee will assign a unique identifier as a token to each user. User are multiple such as user 1, user 2.....user n (assume 3 here). They are having different roles. Some are creator who will securely store data in cloud. Some are readers, and some are writers. They are differentiated based on their privileges to access the data. Writer is having access to modify the data, but reader can only read the data. It is a decentralized scheme because there are multiple KDCs, which will generate key to user for accessing the data in cloud.

When user wants to upload a file to the cloud he first needs to register with trustee then trustee assigns a token to him which is required for further process. User will present his token to one or more KDCs. After verification at KDC , If user is valid then KDC will assign the keys for encryption and decryption to user depend on the access The data will encrypted under some access policy and securely stored in cloud. Here for file storage securely in cloud we are using homomorphic encryption technique that is paillier encryption [3]. Computationally this scheme is very strong. We are generating hash signature using the SHA-1 before storing the data on cloud at user side. When user want to upload the data on cloud, system checks the hash generated by cloud and user if it matches then the copy of file will be stored on cloud securely. When other users are interested in reading or writing the files, the access will provided only to users satisfying access policy. Similar to upload file process, the user send request for downloading a file and if satisfy the access policy he can decrypt the file.

ALGORITHM

For authentication we are generating signature hash value using SHA-1 It works as follows.

Step-1 Padding

- Pad the given message with a single one followed by number of zeroes until the final block has 448 bits.
- Append the size of the original message as an unsigned 64 bit integer.

Step- 2 Initialize the 5 hash blocks (h0, h1, h2, h3, h4) to the specific constant values defined in the SHA-11 standard.

Step 3 Hash (for each 512-bit Block)

1. Allocate an 80 word array for the message.
    - Set the first 16 words to be the 512-bit block split into 16 words.
    - The rest of the words are generated using the following algorithm
        - word[i-3] XOR word[i-8] XOR word[i-14] XOR word[i-16] then rotated 1 bit to the left
2. Loop 80 times doing the following as shown in fig 2
    - Calculate SHA function () and the constant K (these are based on the current round number.
    - e=d
    - d=c
    - c=b (rotated left 30)
    - b=a
    - a = a (rotated left 5) + SHA function() + e + k + word[i]
3. Add a, bcc'd and e to the hash output.

Step 4 Output the concatenation (h0,h1,h2,h3,h4) which is the message digest.

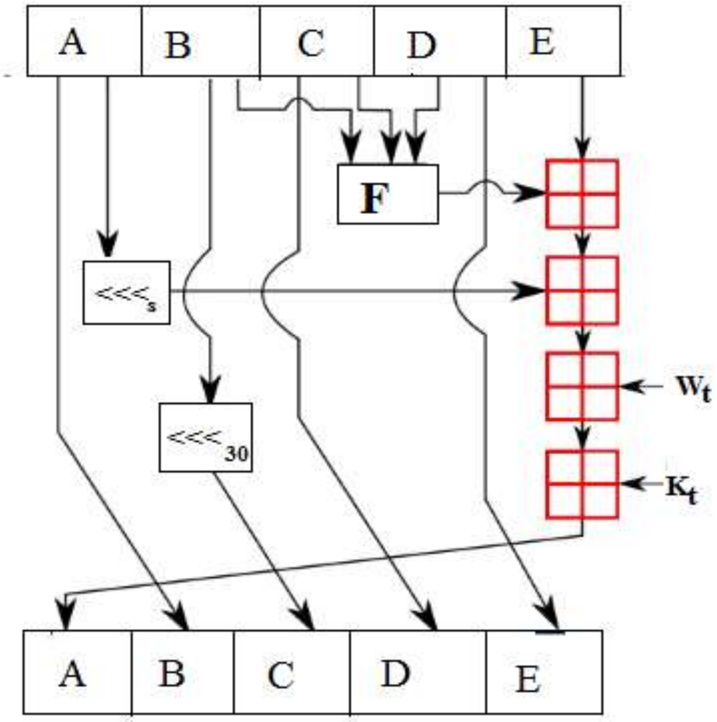

Figure 2: SHA-1

RESULTS

The efficiency of the system is analysed in terms of encryption and decryption time requirement. Here we have compared this system with symmetric key based system (3DES). Files with different sizes were given as a input and encrypted, the result show this system requires less time in milliseconds than symmetric algorithm (3DES).

Table 1. Comparison with DES

Input File Size(kb)	Encryption time		Decryption time	
	3DES(Symmetric)	Paillier(Asymmetric)	3 DES(Symmetric)	Paillier(Asymmetric)
4	3.60	2.70	3.40	2.60
6	7.30	3.30	7.00	3.10
8	8.40	3.50	8.15	3.25
12	13.70	8.10	13.10	7.90

CONCLUSION

We proposed a secure cloud storage scheme that addresses security, privacy, authentication issues in the cloud simultaneously. The key distribution centers are multiple which manages keys so this is distributed in nature. Here used homomorphic encryption which is computationally complex so unauthorized user can not access data.

FUTURE SCOPE

One limitation of this system is that data of type audio video requires more time. In future we will try to implement a method that will reduce this time.

REFERENCES

1. 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.

2. Young-Gi Min, Hyo-Jin Shin, Young-Hwan Bang, “Cloud Computing Security Issues and Access Control Solutions”, Journal of Security Engineering, 2012.
3. C. Gentry, “A Fully Homomorphic Encryption Scheme,” PhD dissertation, Stanford Univ., http://www.crypto.stanford.edu/ craig, 2009.
4. A. Sahai and B. Waters, “Fuzzy Identity-Based Encryption,” Proc. Ann. Int’l Conf. Advances in Cryptology (EUROCRYPT), pp. 457-473, 2005.
5. 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.
6. J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-Policy Attribute-Based Encryption,” Proc. IEEE Symp. Security and Privacy, pp. 321-334, 2007.
7. M. Chase, “Multi-Authority Attribute Based Encryption,” Proc. Fourth Conf. Theory of Cryptography (TCC), pp. 515-534, 2007.
8. V. Bozovic1, D. Socek, R. Steinwandt, and V. Villanyi, “Multi-authority attribute based encryption with honest-but-curious central authority”.
9. 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.
10. 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.