IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 1, March, 2013
ISSN: 2320 - 8791
www.ijreat.org

Trilinear Entity Based Encryption to Provide Flexible and Fine-Grained Access in Cloud Computing

B.Jagadeesh [1], G.ArunKumar [2]and S.Archana [3]

[1,]**Asst Professor, Department of computer Science and Engineering,**
Anand Institute of Higher Technology,
Chennai-603103.

[2]**. Asst.professor, Department of Information Technology,**
Anand Institute of Higher Technology,
Chennai-603103.

[3]**. P.G Student, Department of computer Science and Engineering ,**
Anand Institute of Higher Technology,
Chennai-603103.

Abstract

In the era of technology data sharing and resource sharing has became a basic aspect which lead to the introduction of cloud. They have improved security and privacy for out sourced data's by the use of a concept known as ABE(Attribute based encryption) however the suffer from inflexibility and scalability regarding access control of outsourced data of cloud computing . in order to realize them and improve security trilinear encryption concept known as trilinear entity based encryption is used . we implement our scheme and show that it is both efficient and flexible in dealing with access control for outsourced data in cloud computing with comprehensive experiments.

1. INTRODUCTION

CLOUD computing is a new computing paradigm that is built on virtualization, parallel and distributed computing, utility computing, and service-oriented architecture. In the last several years, cloud computing has emerged as one of the most influential paradigms in the IT industry, and has attracted extensive attention from both academia and industry. Cloud computing holds the promise of providing computing

Although the great benefits brought by cloud computing paradigm are exciting for IT companies, academic researchers, and potential cloud users, security problems in cloud computing become serious obstacles which, without being appropriately addressed, will prevent cloud computing extensive applications and usage in the future Data confidentiality is not the only security requirement.

Flexible and fine-grained access control is also strongly desired in the service-oriented cloud computing model. A health-care information system on a cloud is required to restrict access of protected medical records to eligible

doctors and a customer relation management system running on a cloud may allow access of customer information to high-level executives of the company only.

The following are the procedures involved in the project as the data which is subjected to be stored on cloud is encrypted based on trilinear encryption format namely ,first he user name is converted into binary format and then it is considered as public key ,then secondly the master key is created by performing an AND operation of the binary values of the data's name and its size and thirdly the secret key for encryption is created by OR operation of the binary format of public and master key and then reversed using NOT operation .and the rest of the paper will illustrate the merits and overview of the system and procedures involved to implement them with diagrammatic representations and algorithms.

2. RELATED WORK

In this section, we review the notion of attribute-based encryption (ABE), and provide a brief overview of the

ASBE scheme by Bobba et al. After that, we examine existing access control schemes based on ABE.

2.1 Attribute-Based Encryption

The notion of ABE was first introduced by Sahai and Waters[1] as a new method for fuzzy identity-based encryption. The primary drawback of the scheme in [1] is that its threshold semantics lacks expressibility. Several efforts followed in the literature to try to solve the expressibility problem. In the ABE scheme, ciphertexts are not encrypted to one particular user as in traditional public key cryptography. Rather, both ciphertexts and users' decryption keys are associated with a set of attributes or a policy over attributes. A user is able to decrypt a ciphertext only if there is a match between his decryption key and the ciphertext. ABE schemes are classified into key-policy attribute- based encryption (KP-ABE) and ciphertext-policy attribute-based encryption (CP-ABE)[3], depending how attributes and policy are associated with ciphertexts and users' decryption keys.

2.2 Access Control Solutions for Cloud Computing

The traditional method to protect sensitive data outsourced to third parties is to store encrypted data on servers, while the decryption keys are disclosed to authorized users only. However, there are several drawbacks about this trivial solution. First of all, such a solution requires an efficient key management mechanism to distribute decryption keys to authorized users, which has been proven to be very difficult. Next, this approach lacks scalability and flexibility; as the number of authorized users becomes large, the solution will not be efficient anymore. In case a previously legitimate user needs to be revoked, related data has to be re-encrypted and new keys must be distributed to existing legitimate users again. Last but not least, data owners need to be online all the time so as to encrypt or re-encrypt data and distribute keys to authorize users.

Wang et al. [2] proposed hierarchical attribute-based encryption (HABE) to achieve fine-grained access control in cloud storage services by combining hierarchical identity-based encryption (HIBE) and CP-ABE. This scheme also supports fine-grained access control and fully delegating computation to the cloud providers. However, HABE uses disjunctive normal form policy and assumes all attributes in one conjunctive clause are administrated by the same domain master. Thus the same attribute may be administrated by multiple domain masters according to specific policies, which is difficult to implement in practice. Furthermore, compared with ASBE, this scheme cannot support compound attributes efficiently and does not support multiple value assignments.

3. SYSTEM MODELAND ASSUMPTIONS

3.1. System Model

As depicted in Fig. 1, the cloud computing system underconsideration consists of five types of parties: a cloud serviceprovider, data owners, data consumers, a number of domain authorities, and a trusted authority.The cloud service provider manages a cloud to provide datastorage service. Data owners encrypt their data files and storethem in the cloud for sharing with data consumers. To accessthe shared data files, data consumers download encrypted datafiles of their interest from the cloud and then decrypt them. Eachdata owner/consumer is administrated by a domain authority. A domain authority is managed by its parent domain authority orthe trusted authority. Data owners, data consumers, domain authorities,and the trusted authority are organized in a hierarchical manner as shown in Fig. 1. The trusted authority is the root authority and responsible for managing top-level domain authorities. Each top-level domain authority corresponds to a top-level organization, such as a federated enterprise, while each lower-level domain authority corresponds to a lower-level organization, such as an affiliated company in a federated enterprise.

Data owners/consumers may correspond to employees in an organization. Each domain authority is responsible for managing the domain authorities at the next level or the data owners/consumers in its domain[4].

In our system, neither data owners nor data consumers will be always online. They come online only when necessary, while the cloud service provider, the trusted authority, and domain authorities are always online. The cloud is assumed to have abundant storage capacity and computation power. In addition, we assume that data consumers can access data files for reading only.

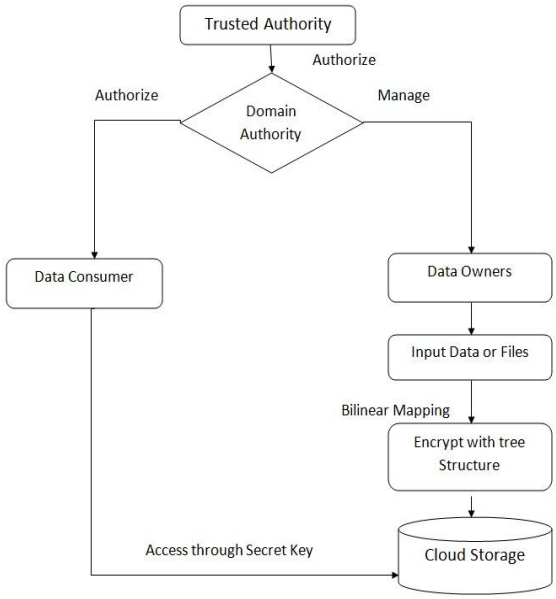

Fig. 1. System model

3.2. Security Model

We assume that the cloud server provider is untrusted in the sense that it may collude with malicious users (short for data owners/data consumers) to harvest file contents stored in the cloud for its own benefit. In the hierarchical structure of the system users given in Fig. 1, each party is associated with a public key and a private key, with the latter being kept secretly by the party[4]. The trusted authority acts as the root of trust and authorizes the top-level domain authorities. A domain authority is trusted by its subordinate domain authorities or users that it administrates, but may try to get the private keys of users outside its domain. Users may try to access data files either within or outside the scope of their access privileges, so malicious users may collude with each other to get sensitive files beyond their privileges[4]. In addition, we assume that communication channels between all parties are secured using standard security protocols, such as SSL.

4. ADVANTAGE OF SYSTEM AND IMPLEMENTATION

4.1. Advantage of system

More specifically, we associate each data file with a set of attributes, and assign each user an expressive access structure whichis defined over these attributes. To enforce this kind of access control, we utilize KP-ABE to escort data encryption keys of data files. Such construction enables us to immediately enjoy fine-grainedness of access control. However, this construction, if deployed alone, would introduce heavy computation overhead and cumbersome onlineburden towards the data owner, as he is in charge of all the operations of data/user management .

- Low initial capital investment
- Shorter start-up time for new services
- Lower maintenance and operation costs
- Higher utilization through virtualization
- Easier disaster recovery

Specifically, such an issue is mainly caused by the operation of user revocation, which inevitably requires the data owner to re-encrypt all the data files accessible to the leaving user, oreven needs the data owner to stay online to update secret keys for users. To resolve this challenging issue and make the construction suitable for cloud computing, we uniquely combine PRE with KP-ABE and enable the data owner to delegate most of the computation intensive operations to Cloud Servers without disclosing the underlying file contents[3]. Such a construction allows the data owner to control access of his data files with a minimal overhead in terms of computation effort and online time, and thus fits well into the cloud environment. Data confidentiality is also achieved since Cloud Serversare not able to learn the plaintext of any data file in our construction. For further reducing the computation overhead on Cloud Servers andthus saving the data owner's investment, we take advantage of thelazy re-encryption technique and allow Cloud Servers to "aggregate "computation tasks of multiple system operations. As we will discuss in section V-B, the computation complexity on Cloud Servers is eitherproportional to the number of system attributes, or linear to the size of the user access structure/tree, which is independent to the number of users in the system. Scalability is thus achieved. In addition, our construction also protects user access privilege information againstCloud Servers. Accountability of user secret key can also be achieved by using an enhanced scheme of KP-ABE[3]

4.2 .Implementation

Initially a secret key to encrypt the data stored in cloud requires these three phases where the first phase involves the binary conversion of the user name or any other unique data sequel for the public key generation and it is treated as the public key and then the file name and size is converted to their respected binary format and are

formed together based on AND operation and it is considered as the master key ,when it is generated secret key is produced by OR operation of the two key values and then the final key is generated by performing a NOT operation to shift the values to reverse and is used respectively to generate the encryption key for data's that are stored in cloud[3] similar to HASBE.

5. CONCLUSION

In this paper, we introduced the TEBE scheme for realizing scalable, flexible, and fine-grained access control in cloud computing. The TEBE scheme seamlessly incorporates a hierarchical structure of system users by applying a delegation algorithm to ASBE. TEBE not only supports compound attributes due to flexible attribute set combinations, but also achieves efficient user revocation because of multiple value assignments of attributes. Finally, we implemented the proposed scheme, and conducted comprehensive performance analysis and evaluation, which showed its efficiency and advantages over existing schemes.

REFERENCES

[1] A. Sahai and B. Waters, "Fuzzy identity based encryption," in Proc. Acvances in Cryptology—Eurocrypt, 2005, vol. 3494, LNCS, pp. 457–473.

[2] G.Wang, Q. Liu, and J.Wu, "Hierachical attibute-based encryption for fine-grained access control in cloud storage services," in Proc. ACM Conf. Computer and Communications Security (ACM CCS), Chicago, IL, 2010.

[3] Zhiguo Wan, Jun'e Liu, and Robert H. Deng, Senior Member, IEEE "Hierachical attibute-based encryption for fine-grained access control in cloud storage services," with Key Laboratory for Information System Security, Ministry of Education, Tsinghua National Laboratory for Information Science and Technology,China ,2012

[4] P. D. McDaniel and A. Prakash, "Methods and limitations of security policy reconciliation," in Proc. IEEE Symp. Security and Privacy, Berkeley, CA, 2002.