ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through
www.ijptonline.com

SPEC- SERIAL PROPERTY BASED ENCRYPTION FOR CLOUD

[1]N.Partheeban, [2]K.Sudharson, [3]P.J.Sathish Kumar

[1]Associate Professor, Department of Information Technology, S.A.Engineering College, Chennai-77.
[2]Assistant Professor, Department of Information Technology, S.A.Engineering College, Chennai-77.
[3]Research Scholar, Department of CSE, Bharath Institute of Higher Education and Research,
BharathUniversity, Chennai.
Email: Knparth78@gmail.com

Received on: 15.10.2016 *Accepted on: 22.11.2016*

Abstract

Cloud computing has emerged as one of the mostinfluential paradigms in the IT industry in recent years. Since this new computing technology requires users to entrust their valuable data to cloud providers, there have been increasing security and privacy concerns on outsourced data. Several schemes employing attribute-based encryption (ABE) have been proposed for access control of outsourced data in cloud computing; However, most of them suffer from inflexibility in implementing complex access control policies. In order to realize scalable, flexible, and fine grained access control of outsourced data in cloud computing, we propose hierarchical attribute-set-based encryption (HASBE) by extending cipher text policy attribute-set-based encryption (ASBE) with a hierarchical structure of users. The proposed scheme not only achieves scalability due to its hierarchical structure, but also inherits flexibility and fine-grained access control in supporting compound attributes of ASBE.

Keywords: Access control; Cloud computing; Data security.

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 as the fifth utility after the other four utilities (water, gas, electricity, and telephone). The benefits of cloud computing include reduced costs and capital expenditures, increased operational efficiencies, scalability, flexibility, immediate time to market, and so on. Different service-oriented cloud computing models have been proposed, including infrastructure as a service (iaas), platform as a service (paas), and software as a service (saas). 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's extensive applications and usage in the future. One of the prominent security concerns is data security and privacy in cloud computing due to its internet-based data storage and management. Cloud users in the first place want to make sure that their data are kept confidential to outsiders, including the cloud provider and their potential competitors. This is the first data security requirement. 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. A new access control scheme employing attributed-based encryption is proposed by yu*et al.* which adopts the so-called key-policy attribute-based encryption (kp-abe) to enforce fine-grained access control. However, this scheme falls short of flexibility in attribute management and lacks scalability in dealing with multiple-levels of attribute authorities we note that in contrast to kp-abe, cipher-text policy abe (cp-abe) turns out to be well suited for access control due to its expressiveness in describing access control policies. In this paper, we propose a hierarchical attribute-set-based encryption (hasbe) scheme for access control in cloud computing. Hasbe extends the cipher-text policy attribute-set-based encryption (cp-asbe, or asbe for short) scheme by bobba*et al.* With a hierarchical structure of system users, so as to achieve scalable, flexible and fine-grained access control.

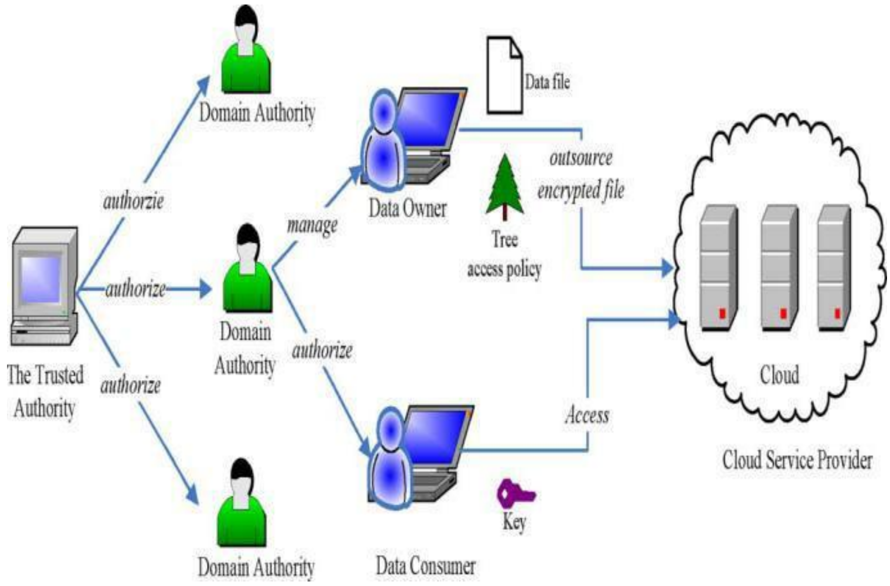
2. Literature Review

A. Existing System: Cloud storage enables networked online storage where data is stored on multiple virtual servers, generally hosted by third parties, rather than being hosted on dedicated servers. Having remote database, there arises security problems. So in order to maintain data integrity, proposed design consists of efficient methods that enable on demand data correctness verification. A traditional cloud security concept ensures the identity based cryptography which results in secured outsourcing of cloud data. HASBE extends the ABSE algorithm with a hierarchical structure to improve scalability and flexibility while at the same time inherits the feature of fine grained access control ABSE cloud system has the computation assigned in a great number of distributed computers, rather than local computer or remote server. Though cloud services are entirely based on distributed computing, broad range of both internal and external threats for data integrity still exist. Thus, distributed protocols for storage correctness assurance will be of most importance in achieving robust and secure cloud storage systems.

B. Limitations of the existing system

Focused on data integrity rather than security propose a hierarchical attribute-set-based encryption (HASBE) scheme for access control in cloud computing. HASBE extends the cipher text-policy attribute- set-based encryption with a hierarchical structure of system users, so as to achieve scalable, flexible and fine-grained access control. In this Hasbe process, the cipher text is encrypted with a tree access policy chosen by an encryptor, while the corresponding decryption key is created with respect to a set of attributes. As long as the set of attributes associated with a decryption key satisfies the tree access policy associated with a given cipher text, the key can be used to decrypt the cipher text. Through Hasbe, we have achieved fine grained access control and secured outsourcing of cloud data in a multiuser environment categorized in a hierarchical manner. No effective data encryption algorithms are 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.

C. Proposed System



In my 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.

3. Methodology

A. Architecture of Proposed System

The cloud computing system under consideration consists of five types of parties: a cloud service provider, data owners, data consumers, a number of domain authorities and a trusted authority. The loud service provider manages a cloud to provide data storage service. Data owners encrypt their data files and store them in the cloud for sharing with data consumers. To access the shared data files, data consumers download encrypted data files of their interest from the cloud and then decrypt them. Each data owner/ consumer is administrated by by a domain authority. A domain authority is managed by its parent domain authority or trusted authority. Data owners, data consumers, domain authorities and the trusted authority are organised in a hierarchical manner.

The trusted authority is the root authority and responsible for managing top level domain authority corresponds to a top level organisation, such asa federated enterprise, which each lower level domain authority corresponds to a lower level organisation, such as an affiliated company in a deployed in existing system. No error recovery algoritms are implemented in existing system.

B. Attribute-Based Encryption

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 ciphertextnly 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), depending how attributes and policy are associated with ciphertexts and users' decryption keys. so in my project I proposed HASBE scheme.

C. HASBE Scheme

The proposed HASBE scheme seamlessly extends the ASBE scheme to handle the hierarchical structure of system users. Recall that our system model consists of a trusted authority, multiple domain authorities, and numerous users corresponding to data owners and data consumers. The trusted authority is responsible for generating and distributing system parameters and root master keys as well as authorizing the top-level domain authorities. A domain authority is responsible for delegating keys to subordinate domain authorities at the next level or users in its domain. Each user in the system is assigned a key structure which specifies the attributes associated with the user's decryption key. To achieve a flexible, scalable and fine-grained access control in cloud computing by using Hierarchical Attribute Set Based Encryption(HASBE).we compare our scheme with the one proposed by Yu *et al.* on security features in implementing access control for cloud computing.

1) Scalability: We extend ASBE with a hierarchical structure to effectively delegate the trusted authority's private attribute key generation operation to lower-level domain authorities. By doing so, the workload of the trusted root

authority is shifted to lower-level domain authorities, which can provide attribute key generations for end users. Thus, this hierarchical structure achieves great scalability. Yu *etal.*'s scheme, however, only has one authority to deal with key generation, which is not scalable for large-scale cloud computing applications.

2) Flexibility: Compared with Yu *et al.*'s scheme, HASBE organizes user attributes into a recursive set structure and allows users to impose dynamic constraints on how those attributes may be combined to satisfy a policy. So HASBE can support compound attributes and multiple numerical Assignments for a given attribute conveniently.

3) Fine-grained access control: Based on HASBE, our scheme can easily achieve fine-grained access control. A data owner can define and enforce expressive and flexible access policy for data files

D. Modules

1. Root and Domain Authority

Root authority holds the topmost priority in the secure cloud storage and access system and Administrate the domain authority. Domain authorities administrate the data owner who owns the data in the cloud.

2. Bilinear Mapping

Attribute Based Encryption (ABE) is preceded by bilinear mapping of attribute information of data owner and the data to be stored in the cloud. It can achieved by multiplicative factors of both logical AND and XOR operations.

3. Master and Secret key

Master key is generated by doing the logical AND operation and given attributes of data owner. Using the master key public key is generated and secret key is generated by doing the logical XOR operation. The secured secret key is generated by ABE.

4. Secure cloud storage

The security is applied in the data owner's file and those files are stored in the cloud servers.

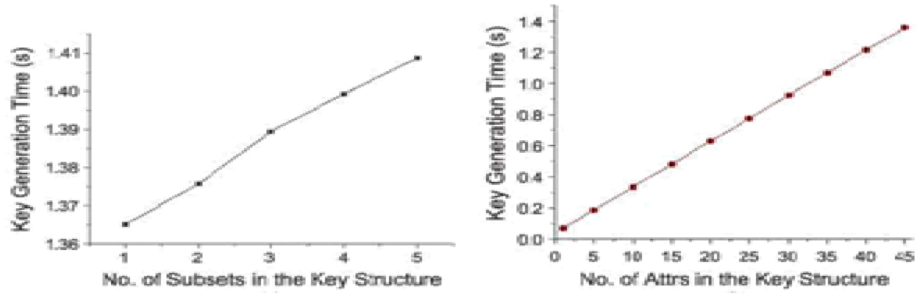
For this crypto process is applied such as blow fish algorithm is used for both encryption and decryption.

5. Secure data Retrieval

The data to be retrieved very secure and decryption is performing using secret key.

4. Performance Analysis

First analyze theoretic computation complexity of the proposed scheme in each operation. Then we implement an HASBE toolkit based on the toolkit developed for CP-ABE and conduct a series of experiments to evaluate performance of our proposed scheme. We analyze the computation complexity for each system operation in our scheme as follows.



A. System Setup

When the system is set up, the trusted authority selects a bilinear group and some random numbers.When PK and MK_0 are generated, there will be several exponentiation operations. So the computation complexity of System Setup is O(1) .

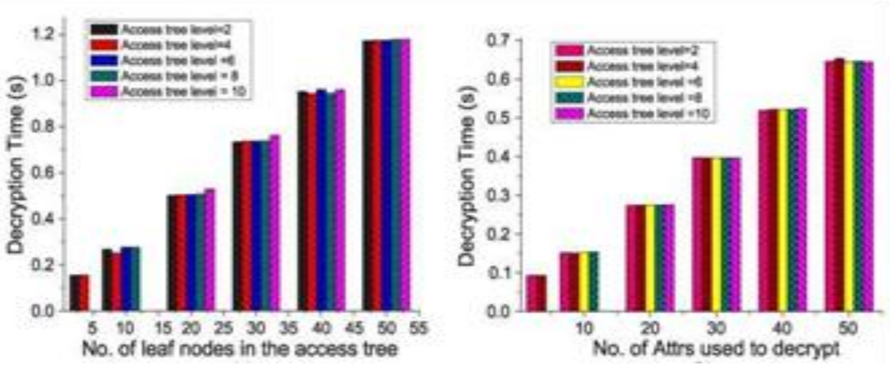
B. Top-Level Domain Authority Grant

This operation is performed by the trusted authority. The master key of a domain authority is

in the form of,MK_0=(A,D,$D_{i,j}$,$D'_{i,j}$ for $a_{i,j}$ ϵ A,E_i for AϵA$_i$),where A is the key structure associated with anew domain authority,A is the set of A_i . Let N be the number of attributes in A , and M be the number of sets in A . Then the computation of MK_i consists of two exponentiations for each attribute in A , and one exponentiations for every set in A . The computation complexity of *Top-Level Domain* Authority Grant operation is O(2N + M) .

C. New User/Domain Authority Grant

In this operation, a new user or new domain authority is associated with an attribute set, which is the set of that of the upper level domain authority. The main computation overhead of this operation is re-randomizing the key. The computation complexity is O(2N + M) , where N is the number of attributes in the set of the new user or domain authority, and is the M number of sets in A.



D. New File Creation

In this operation, the data owner needs to encrypt a data file using the symmetric key DEK and then encrypt DEK using HASBE. The complexity of encrypting the data file with DEK depends on the size of the data file and the underlying symmetric key encryption algorithm. Encrypting DEK with a tree access structure T consists of two exponentiations per leaf node in T and one exponentiation per translating node in T . So the computation complexity of New File Creation is O(2|Y| + |X|) ,where Y denotes the leaf nodes of T and X denotes the translating nodes of T.

E. File Access

In this operation, we discuss the decrypting operation of encrypted data files. A user first obtains DEKS with the Decrypt algorithm and then decrypt data files using DEK_S. We will discuss the computation complexity of the Decrypt algorithm. The cost of decrypting a cipher-text varies depending on the key used for decryption. Even for a given key, the way to satisfy the associated access tree may be various.

The Decrypt algorithm consists of two pairing operations for every leaf node used to satisfy the tree, one pairing for each translating node on the path from the leaf node used to the root and one exponentiation for each node on the path from the leaf node to the root. So the computation complexity varies depending on the access tree and key structure. It should be noted that the decryption is performed at the data consumers; hence, its computation complexity has little impact on the scalability of the overall system

F. File Deletion

This operation is executed at the request of a data owner. If the cloud can verify the requestor is the owner of the file, the cloud deletes the data file. So the computation complexity is O(1). Computation complexity of each system operation is shown in Table I, in which N denotes the number of attributes in the key structure, Y is the set of leaf nodes of the access tree or policy tree, and X is the set of translating nodes of the policy tree.

5. Conclusion And Future Work

I introduced the HASBE scheme for realizing scalable, flexible, and fine-grained access control in cloud computing. The HASBE scheme seamlessly incorporates a hierarchical structure of system users by applying a delegation algorithm to ASBE. HASBE not only supports compound attributes due to flexible attribute set combinations, but also achieves efficient user revocation because of multiple value assignments of attributes. We formally proved the security of HASBE based on the security of CP-ABE by Bethencourt et al.. Finally, I implemented the proposed scheme, and conducted comprehensive performance analysis and evaluation, which showed its efficiency and advantages over existing schemes.

Handling Multiple Concurrent Tasks

Such a technique supports the aggregation of multiple signatures by distinct signers on distinct messages into a single signature and thus allows efficient verification for the authenticity of all messages.

Usage of High security cryptographic algorithms

Using Blowfish algorithm, which is 448 bits key length results in higher security, rather than using traditional DES and AES algorithms are smaller in key sizes results in lesser security.

Reducing Storage Overhead

By means of de-duplication concept, we can reduce storage overhead problem that reduces the problem of redundant data of multiple users being stored in cloud storage.

References

1. Zhiguo Wan, Jun'e Liu, and Robert H. Deng, *SeniorMember, IEEE* ―HASBE: A Hierarchical Attribute-BasedSolution for Flexible and Scalable Access Control in Cloud Computing‖ Ieee Transactions On Information Forensics And Security, Vol. 7, No. 2, April 2012.
2. J. Li, N. Li, and W. H. Winsborough, ―Automated trust negotiation using cryptographic credentials,‖ in Proc.ACM Conf. Computer and Communications Security (CCS), Alexandria, VA, 2005.
3. V. Goyal, O. Pandey, A. Sahai, and B.Waters, ―Attibute-based encryption for fine-grained access control of encrypted data,‖ in Proc. ACM Conf. Computer and Communications Security (ACM CCS), Alexandria, VA, 2006.
4. A. Sahai and B. Waters, ―Fuzzy identity based encryption,‖ in Proc. Acvances in Cryptology—Eurocrypt, 2005, vol. 3494, LNCS, pp. 457–473.
5. S. Yu, C. Wang, K. Ren, and W. Lou, ―Achiving secure, scalable, and fine-grained data access control in cloud computing,‖ in *Proc. IEEE INFOCOM 2010*, 2010, pp. 534–542.
6. T. Yu and M. Winslett, ―A unified scheme for resource protection in automated trust negotiation,‖ in *Proc. IEEESymp. Security and Privacy*, Berkeley, CA, 2003.
7. P. D. McDaniel and A. Prakash, ―Methods and limitations of security policy reconciliation,‖ in *Proc. IEEE Symp.Security and Privacy*, Berkeley, CA, 2002.

8. R. Martin, ―IBM brings cloud computing to earth with massive new data centers,‖*InformationWeek* Aug. 2008 [Online]. Available: http://www.informationweek.com/news/hardware/data_ce nters/209901523.
9. K. Barlow and J. Lane, ―Like technology from an advanced alien culture: Google apps for education at ASU,‖ in *Proc. ACM SIGUCCS User Services Conf.*, Orlando, FL, 2007.
10. R. Buyya, C. ShinYeo, J. Broberg, and I. Brandic,―Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility,‖ *Future Generation Comput. Syst.*, vol. 25, pp.599–616, 2009.
11. K. Barlow and J. Lane, ―Like technology from an advanced alien culture: Google apps for education at ASU,‖ in Proc. ACM SIGUCCS User Services Conf., Orlando, FL, 2007.
12. B. Barbara, ―Salesforce.com: Raising the level of networking,‖ Inf. Today, vol. 27, pp. 45–45, 2010.
13. D. E. Bell and L. J. LaPadula, Secure Computer Systems: Unified Ex-position and Multics Interpretation The MITRE Corporation, Tech.Rep., 1976.
14. H. Harney, A. Colgrove, and P. D. McDaniel, ―Principles of policy in secure groups,‖ in Proc. NDSS, San Diego,CA, 2001.