International Journal of Engineering (IJE)
Singaporean Journal of Scientific Research(SJSR)
Vol 6.No.2 2014 Pp. 126-131
available at:www.iaaet.org/sjsr
Paper Received :10-03-2014
Paper Accepted:20-03-2014
Paper Reviewed by: 1Prof.Dr. Somesh2. Chai Cheng Yue
Editor : Dr. Binod Kumar

SECURE, SCALABLE AND FINE GRAINED DATA ACCESS CONTROL USING HASBE

Rizwanullah.S and M.Dilsath Fathima
Wireless & Network Security
Vel Tech Technical University
Chennai

ABSTRACT

Cloud computing has emerged as one of the most influential 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, in this paper, I 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. In addition, HASBE employs multiple value assignments for access expiration time to deal with user revocation more efficiently than existing schemes. I implement this 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 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). Numerous commercial cloud computing systems have been built at different levels, e.g., Amazon's EC2 , Amazon's S3, and IBM's Blue Cloud are IaaS systems, while Google App Engine and Yahoo Pig are representative PaaS systems, and Google's Apps and Salesforce's Customer Relation Management (CRM) System belong to SaaS systems.

With these cloud computing systems, on one hand, enterprise users no longer need to invest in hardware/software systems or hire IT professionals to maintain these IT systems, thus they save cost on IT infrastructure and human resources; on the other hand, computing utilities provided by cloud computing are being offered at a relatively low price in a pay-as-you-use style. For example, Amazon's S3 data storage service with 99.99% durability charges only $0.06 to $0.15 per gigabyte-month, while traditional storage cost ranges from $1.00 to $3.50.

2. SYSTEM DESCRIPTION

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. One of the prominent security concerns is data security and privacy in cloud computing due to its Internet- based data storage and management. In cloud computing, users have to give up their data to the cloud service provider for storage and business operations, while the cloud service provider is usually a commercial enterprise which cannot be totally trusted.

Data represents an extremely important asset for any organization, and enterprise users will face serious consequences if its confidential data is disclosed to their business competitors or the public. Thus, 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. In these cases, access control of sensitive data is either required by legislation (e.g., HIPAA) or company regulations. Access control is a classic security topic which dates back to the 1960s or early 1970s, and various access control models have been proposed since then. Among them, Bell-La Padula (BLP) and BiBa are two famous security models. To achieve flexible and fine-grained access control, a number of schemes have been proposed more recently.

Unfortunately, these schemes are only applicable to systems in which data owners and the service providers are within the same trusted domain. Since data owners and service providers are usually not in the same trusted domain in cloud computing, 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.

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 project, 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.

The contribution of the project is multifold. First, we show how HASBE extends the ASBE algorithm with a hierarchical structure to improve scalability and flexibility while at the same time inherits the feature of fine-grained access control of ASBE. Second, we demonstrate how to implement a full-fledged access control scheme for cloud computing based on HASBE. The scheme provides full support for hierarchical user grant, file creation, file deletion, and user revocation in cloud computing. Third, we formally prove the security of the proposed scheme based on the security of the CP-ABE scheme by Bethen court et al. And analyze its performance in terms of computational overhead. Lastly, we implement HASBE and conduct comprehensive experiments for performance evaluation, and our experiments demonstrate that HASBE has satisfactory performance.

The above desirable feature and the recursive key structure is implemented by four algorithms, Setup, Key Gen, Encrypt, and Decrypt:

Setup (d). Here d is the depth of key structure. Take as input a depth parameter d. It outputs a public key PK and master secret key MK.

Key Gen(MK,u,A). Take as input the master secret key MK , the identity of user u , and a key structure A. It outputs a secret key SK_u for user u .

Encrypt (PK,M,T). Take as input the public key PK , a message M, and an access tree T. It outputs a ciphertext CT.

Decrypt (CT,SK_u). Take as input a cipher text CT and a secret key SK_u for user u. It outputs a message m. If the key structure A associated with the secret key SK_u satisfies the access tree T, associated with the cipher text CT, then m is the original correct message M. Otherwise, m is null.

SECURITY FEATURES:

The security features in implementing access control for cloud computing are,

1) Scalability: I 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 et al.'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. As illustrated with the example key structure in Fig. 2 and access structure in Fig. 3, HASBE can enforce more complex access policies than Yu et al.'s scheme.

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.

4) Efficient User Revocation: To deal with user revocation in cloud computing, we add an attribute to each user's key and employ multiple value assignments for this attribute. So we can update user's key by simply adding a new expiration value to the existing key. We just require a domain authority to maintain some state information of the user keys and avoid the need to generate and distribute new keys on a frequent basis, which makes our scheme more efficient than existing schemes.

5) Expressiveness: In HASBE, a user's key is associated with a set of attributes, so HASBE is conceptually closer to traditional access control methods such as Role-Based Access Control (RBAC). Thus, it is more natural to apply HASBE, instead of KP-ABE, to enforce access control.

3. RELATED WORKS

This literature survey is based upon the reference of several papers related to this project. The history of this project was started from paper [1], in this paper, we define Cloud computing and provide the architecture for creating Clouds with market-oriented resource allocation by leveraging technologies such as Virtual Machines (VMs).In paper [2], as more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites. One drawback of encrypting data is that it can be selectively shared only at a coarse-grained level (i.e., giving another party your private key). We develop a new cryptosystem for fine-grained sharing of encrypted data that we call Key-Policy Attribute-Based Encryption (KP-ABE).

In the paper [3], we present a system for realizing complex access control on encrypted data that we call cipher text-policy attribute-based encryption. By using our techniques encrypted data can be kept confidential even if the storage server is un trusted; moreover, our methods are secure against collusion attacks. In paper [4], Cipher text is associated with set of attributes where the decryption key is associated with tree access structure. Interior nodes of the access tree are threshold gates and leaf nodes are associated with attributes. User secret key is defined to reflect the access structure so that the user is able to decrypt a ciphertext if and only if the data attributes satisfy his access structure.

The existing system focuses on despite the tremendous benefits, outsourcing computation to the commercial public cloud is also depriving customers' direct control over the systems that consume and produce their data during the computation, which inevitably brings in new security concerns and challenges towards this promising computing model.

On the one hand, the outsourced computation workloads often contain sensitive information, such as the business financial records, proprietary research data, or personally identifiable health information etc. Thus, the cloud is intrinsically *not secure* from the viewpoint of customers.

Without providing a mechanism for secure computation outsourcing, i.e., to protect the sensitive input and output information of the workloads and to validate the integrity of the computation result, it would be hard to expect cloud customers to turn over control of their workloads from local machines to cloud solely based on its economic savings and resource flexibility.

For practical consideration, such a design should further ensure that customers perform fewer amounts of operations following the mechanism than completing the computations by themselves directly. Otherwise, there is no point for customers to seek help from cloud. Recent researches in both the cryptography and the theoretical computer science communities have made steady advances in "secure outsourcing expensive computations".

4. PROPOSED WORK

Schemes employing attribute-based encryption (ABE) have been proposed for access control of outsourced data in cloud computing On the one hand, the outsourced computation workloads often contain sensitive information, such as the business financial records, proprietary research data, or personally identifiable health information etc. Formal Security Model: Before giving a formal proof for the proposed scheme. Combat against unauthorized information leakage, sensitive data have to be encrypted before outsourcing so as to provide end to- end data confidentiality assurance in the cloud and beyond.

However, ordinary data encryption techniques in essence prevent cloud from performing any meaningful operation of the underlying plaintext data, making the computation over encrypted data a very hard problem. On the other hand, the operational details inside the cloud are not transparent enough to customers. As a result, there do exist various motivations for cloud server to behave unfaithfully and to return incorrect results, i.e., they may behave beyond the classical semi honest model.

Fully homomorphic encryption (FHE) scheme, a general result of secure computation outsourcing has been shown viable in theory, where the computation is represented by an encrypted combinational Boolean circuit that allows to be evaluated with encrypted private inputs. The advantages of this system are:-

- 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.

5. SYSTEM IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Implementation is the process of converting a new system design into operation. It is the phase that focuses on user training, site preparation and file conversion for installing a candidate system. The important factor that should be considered here is that the conversion should not disrupt the functioning of the organization.

MODULES:

A Hierarchical Attribute Set Based Encryption scheme seamlessly extends the ASBE scheme to handle the hierarchical structure of system users. The Scheme provides full support for

- Hierarchical user grant.
- File creation, file access and file deletion.
- User revocation in cloud computing.

o The Modules are as follows
  - Trusted Authority
  - Domain Authority
  - Data Owner
  - Data Consumer

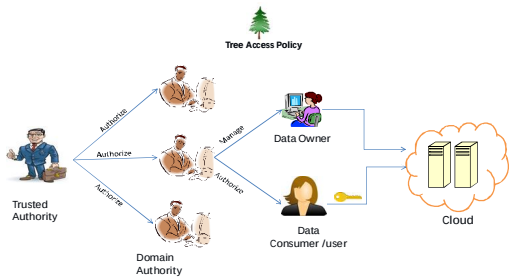

Figure 1:- System Architecture

MODULE DESCRIPTION:

1. TRUSTED AUTHORITY

This is a module mainly designed to authorize the domain authority. Here the domain authority can have the accessibility authority after the registration by the trusted authority.

2. DOMAIN AUTHORITY

This is a module designed to authorize and manage the data owner and data consumer. The data owner and data consumer registration is done here. The domain authority can view the files uploaded details by the data owner and also generate the secret key for the data file requested by the consumer. The domain authority cannot access the data file uploaded by the data owner.

3. DATA OWNER

This is a module for the user who uploading the data files. The registered data owner can upload file, view the uploaded files, generate public key and master key and also for deleting the data files.

4. DATA CONSUMER

This module is for the data consumer who in need of the file to download. The registered data consumer allowed viewing the uploaded file list and sending request to the domain authority requesting the secret key to download the requested file. Using the secret key the data consumer can download the requested file. The data consumer can send request for a particular file only once.

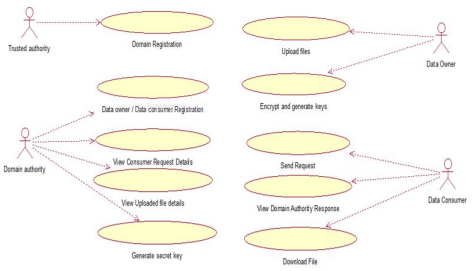

Figure 2:- U case diagram

6. PERFORMANCE RESULTS

The main objective of testing is to uncover errors from the system. For the uncovering process we have to give proper input data to the system. So we should have more conscious to give input data. It is important to give correct inputs to efficient testing.

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus the system testing is a confirmation that all is correct and an opportunity to show the user that the system works. Inadequate Testing or non-testing leads to errors that may apr few months later.

This will create two problems, Time delay between the cause and appearance of the problem. The Effect of the system errors on files and records within the system. The purpose of the system testing is to consider all the likely variations to which it will be suggested and push the system to its limits.

The testing process focuses on logical intervals of the software ensuring that all the statements have been tested and on the function intervals (i.e.,) conducting tests to uncover errors and ensure that defined inputs will produce actual results that agree with the required results. Testing has to be done using the two common steps Unit testing and Integration testing. The obtained outcomes are,

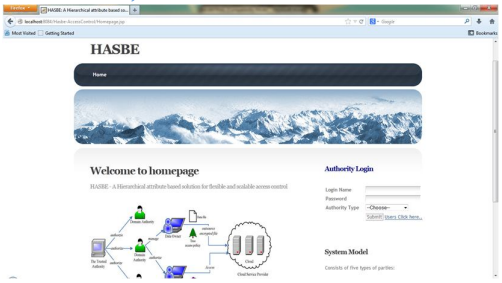

Figure 3: Homepage

7. CONCLUSION & FUTURE ENHANCEMENT

In this project HASBE, we introduced the scheme for realizing scalable, flexible, and fine-grained access control in cloud computing. The HASBE scheme seamlessly incorporates a hierarchical structure of system users. HASBE not only supports compound attributes due to flexible attribute set combinations. We formally proved the security of HASBE based on the security of CP-ABE. Finally, we implemented the proposed scheme, and conducted comprehensive performance analysis and evaluation, which showed its efficiency and advantages over existing schemes.

The project has covered almost all the requirements. Further requirements and improvements can easily be done since the coding is mainly structured or modular in nature. Improvements can be appended by changing the existing modules or adding new modules.

Make a log so that we can examine the performance of this project by watching the user activities. It is possible to request the domain authority for creating an account using an fake name / details, so I suggest that once the user wants to create an account / download the file, both the data owner and the domain authority should get an alert, if they both accept him means the user will be considered as an authorized user.

REFERENCES

1. 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. Amazon Elastic Compute Cloud (Amazon EC2) [Online]. Available: http://aws.amazon.com/ec2/
Amazon Web Services (AWS) [Online]. Available:https://s3.amazonaws. com/

2. 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_centers/209901523 Google App Engine [Online]. Available: http://code.google.com/appengine/

3. 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.

4. B. Barbara, “Salesforce.com: Raising the level of networking,” Inf. Today, vol. 27, pp. 45–45, 2010.

5. J. Bell, Hosting EnterpriseData in the Cloud—Part 9: InvestmentValue Zetta, Tech. Rep., 2010.

6. A.Ross, "Technical perspective: A chilly sense of security," Commun. ACM, vol. 52, pp. 90–90, 2009.

7. D. E. Bell and L. J. LaPadula, Secure Computer Systems: Unified Exposition and Multics Interpretation The MITRE Corporation, Tech. Rep., 1976.

8. K. J. Biba, Integrity Considerations for Secure Computer Sytems The MITRE Corporation, Tech. Rep., 1977.