OPEN ACCESS INTERNATIONAL JOURNAL OF SCIENCE & ENGINEERING

SURVEY ON AUDITABLE ACCESS CONTROL SYSTEMS FOR PUBLIC SHARED CLOUD STORAGE

Miss. Manasi Shet, Prof. Dr. S. N. Kini

Department of Computer Engineering, JSPM's Jayawantrao Sawant College of Engineering Hadapsar, Pune-411028

Abstract: **In public cloud storage system protecting the data and controlling the data access is a challenging issue. Cipher text Policy Attribute-Based Encryption (CP-ABE) has been adopted as a promising technique to provide flexible, fine-grained and secure data access control for cloud storage with honest-but-curious cloud servers. However numerous works have been proposed using CP-ABE scheme, in which the single attribute authority must execute the time-consuming user legitimacy verification and secret key distribution and hence it results in a single-point performance bottleneck when a CP-ABE scheme is adopted in a large-scale cloud storage system. Clients may be stuck in the waiting line for a long stretch to get their mystery keys, which results in low-efficiency of the framework. Even though the multi authority access control plans have been proposed, these plans still cannot conquer the disadvantages of single-point bottleneck and low efficiency; because of the way that each of the authority still autonomously deals with a disjoint characteristic set. In this work, it has been proposed a novel heterogeneous framework to remove the problem of single point performance bottleneck and provide a more efficient access control scheme with an auditing mechanism. This framework employs multiple attribute authorities to share the load of user legitimacy verification. Meanwhile, in this scheme, a CA (Central Authority) is introduced to generate secret keys for legitimacy verified users and each of the authorities in our scheme manages the whole attribute set individually. This system makes performance improvement in key generation and also guarantees security requirement**

Keywords: *Cloud storage, Access control, Auditing, CPABE.*

I INTRODUCTION

Cloud computing has drawn extensive attentions from both academic and industry to satisfy the requirement of data storage and high performance computations. Cloud storage is an important service of cloud computing which provides services for data owners to outsource data to store in cloud via Internet. Benefits of using cloud storage include greater accessibility, higher reliability, rapid deployment and stronger protection, to name just a few. Despite the mentioned benefits, this paradigm also brings forth new challenges on data access control, which is a critical issue to ensure data security. Since cloud storage is operated by cloud service providers, who are usually outside the trusted domain of data owners, the traditional access control methods in the Client/Server model are not suitable in cloud storage environment. The data access control in cloud storage environment has thus become a challenging issue. To address the issue of data access control in cloud storage, there have been quite a few schemes proposed, among which Cipher text Policy Attribute- Based Encryption(CP-ABE) is regarded as one of the most promising techniques. A salient feature of CP-ABE is that it grants data owners direct control power based on access policies, to provide flexible, fine grained and secure access control for cloud storage systems.

In CP-ABE schemes, the access control is achieved by using cryptography, here an owners data is encrypted with an access structure over attributes, and a users secret key is labeled with his/her own attributes. Only if the attributes associated with the users secret key satisfy the access structure, can the user decrypt the corresponding cipher-text to obtain the plaintext. So far, the CP-ABE based access control schemes for cloud storage have been developed into two complementary categories, namely, single-authority scenario and multi authority scenario. In most existing CP-ABE schemes there is only one authority responsible for attribute management and key distribution. This only-one-authority scenario can bring a single-point bottleneck on both security and performance. Once the authority is compromised, an adversary can easily obtain the only-one-authorities master key, and then he/she can generate private keys of any attribute subset to decrypt the specific encrypted data. Moreover, once the only-one-authority is crashed, the system completely cannot work well. Therefore, these CP-ABE schemes are still far from being widely used for access control in public cloud storage. Although some multi-authority CP-ABE schemes have been proposed, they still cannot deal with the problem of single-point bottleneck on both security and performance mentioned above. In these multi-authority CP-ABE schemes, the whole attribute set is divided into multiple disjoint subsets and each attribute subset is still maintained by only one authority.

A straightforward idea to remove the single-point bottleneck is to allow multiple authorities to jointly manage the universal attribute set, in such a way that each of them is able to distribute secret keys to users independently. In this work ,it has been proposed a novel access control heterogeneous framework to address the low efficiency and single-point performance bottleneck for public cloud storage. It proposes a robust and efficient heterogeneous framework with single CA(Central Authority)and multiple AAs (Attribute Authorities) for public cloud storage. The heavy load of user legitimacy verification is shared by multiple AAs, each of which manages the universal attribute set and is able to independently complete the user legitimacy verification, while CA is only responsible for computational tasks which generate secret keys for legitimacy verified users. To enhance security, we also propose an auditing mechanism to detect which AA (Attribute Authority) has incorrectly or maliciously performed the legitimacy verification procedure.

II RELATED WORK

1) Ciphertext-Policy Attribute-Based Encryption (CP-ABE)

Even though the definitions and constructions of different CPABE schemes are not always accurate, the uses of the access structure in Encrypt and Decrypt algorithms are nearly the same. Here we adopt the definition and construction from [6, 10]. A CP-ABE scheme consists of four algorithms: Setup, Encrypt, Key Generation (KeyGen), and Decrypt.

Setup(λ, U) → (PK,MSK). The setup algorithm takes the security parameter λ and the attribute universe description Uas the input. It outputs the public parameters PK and a master secret key MSK.

Encrypt(PK,M,A) → CT. The encryption algorithm takes the public parameters PK, a message M, and an access structure A as input. The algorithm will encrypt M and produce a ciphertext CT such that only a user whose attributes satisfies the access structure will be able to decrypt the message. We will assume that the ciphertext implicitly contains A.

KeyGen(MSK, S) → SK. The key generation algorithm takes the master secret key MSK and a set of attributes S as input. It outputs a secret key SK.

Decrypt(PK,CT, SK)→M. The decryption algorithm takes the public parameters PK, a ciphertext CT which contains an access policy A, and a secret key SK as input, where SK isa secret key for a set S of attributes. If the set S of attributes satisfies the access structure A, the algorithm will decrypt the ciphertext and return a message M.

2)PRSE(Personalized multi-keyword Ranked Search over Encrypted data) Framework

In Cloud computing searchable encryption is a challenging task. However, most of the existing works follow the model of “one size fits all” and ignore personalized search over outsourced encrypted data. So PRSE framework solves the problem of personalized multi-keyword ranked search over encrypted data by preserving security of the system in cloud computing. This framework builds user interest model for every user with the help of semantic ontology WordNet by analysis user’s search history and by adopting a scoring mechanism to express user interest smartly. This framework supports both personalized multi-keyword ranking search and query extension.[1]

This framework involves three entities: the data owner (owner), the data user (user) and the cloud server (server). There exists a user interest model stored in the user side. User’s interest model is built upon user’s search history since long time. Using WordNet, it records access frequency of both query keywords and their related keywords. Different access frequency of keywords as different priority reflects different importance of keywords with respect to user’s interest. In order to start with search for files of interest, data user has to produce a search request first. And then query reformulation will be carried out by user interest model which achieves user keyword priority of query terms. After this encrypted search query through search control mechanism will be sent to the cloud server. Upon receiving search query from authorized user, the cloud server will conduct some designated search over the index and returns relevant encrypted documents which have been ranked by some ranking criteria (scoring mechanism) by cloud server. Here cloud server is the single authority who does searching, indexing and ranking of relevant documents and sends back to the user.

3) Content aware search over encrypted data

There are many schemes have been proposed to make encrypted data searchable over cloud based on keywords. However keyword based search cannot fulfill the user intention of search as they do not follow semantic representation of information of users' retrieval. This work proposes a semantic search scheme based on concept hierarchy and semantic relationship between the concepts in the encrypted datasets. This scheme first indexes the documents and builds trapdoor based on the concept hierarchy and it is further improved by utilizing tree based index structure for organizing all the documents index vectors [2].

In recent years, the general procedure for searching encrypted data involves five steps: document feature extraction, creating searchable index, generating search trapdoor, searching the index based on trapdoor and return the search results. In this work, concept hierarchy tree is constructed based on related knowledge of domain concepts of outsourced dataset. It extends concept hierarchy to include more semantic relations between concepts. It generates two index vectors for each document, one for matching the concepts in the search request and another one is used to determine whether the attribute value is satisfying the search request. In this system model there are three entities: the data owner, the data user and the cloud server.

The data owner constructs a concept hierarchy based on related knowledge of domain concepts of the documents in datasets to be uploaded, then two index vectors for each document of the dataset are generated based on key concepts of the document and the concept hierarchy and at last searchable index is constructed using all the index vectors. The legal user can search required document from the outsourced encrypted datasets by using encrypted search trapdoor to the cloud server. After receiving the trapdoor the cloud server searches the searchable index for required document and returns those encrypted documents which satisfy the search request.

4) Attribute Based Access Control with Efficient Revocation

Ciphertext-policy attribute-based encryption is a promising cryptographic solution to these issues for enforcing access control policies defined by a data owner on outsourced data several challenges with regard to the attribute and user revocation. access control mechanism using ciphertext-policy attribute-based encryption to enforce access control policies with efficient attribute and user revocation capability. The fine-grained access control can be achieved by dual encryption mechanism which takes advantage of the attribute-based encryption and selective group key distribution in each attribute group [3].

5) LABAC Framework

CP-ABE is a cryptographic technique used for data access control based on user's permanent characteristics. However in some situations the access policy depends on users' both permanent and temporary conditions. In LABAC(Location-aware Attribute Based Access Control) framework users' access policy is determined by their attributes as well as their locations. LABAC integrate CP-ABE with location trapdoors to make up access policies which support fine grained control of their data.

In LABAC sensitive data encrypted under access policies and uploaded to the cloud. The attributes are handled using CP-ABE and location information is introduced using trapdoor inside access policies. Location servers are used to release the trapdoors for users. Location servers provide tokens with which the trapdoors can be released. To decrypt the ciphertext users attribute set should satisfy the access policies formulated by owner and get the tokens from location servers to release the trapdoors. The trapdoor is independent of users attribute sets so users private key is only associated withattribute set and not with temporary locations. Therefore when the location changes, there is no need of revoking and reassigning of users. Thus trapdoor reduces the burden of revoking and reassigning. In LABAC multiple trapdoors are associated with every ciphertext and location trapdoors are set arbitrarily in the access policy along with attribute set. In this system model there six entities: the cloud servers, many data owners, many data users, an attribute authority, and multiple location servers, each with sensors. The data owner encrypts the his/her data under access policy defined by him/her and uploads data to the cloud. Attribute authority is responsible for setting up the system and distribute private keys to users with respect to their attribute keys.

The location servers are the servers located in some particular areas where location information is required to provide access privileges. It helps users to release trapdoors by providing tokens. It states user's location with the help of sensors. Sensors are deployed in the areas around the location servers to help them to authenticate user's location. The data user can download any interested ciphertext from the cloud server and decrypts it as he/she has private key according to his/her attribute set. The cloud server gives platform to store owner's data and share data to user. In this framework attribute authority is a single authority who handles setting up the system, key generation and distribution due to which it leads to single point bottleneck problem even though it provides location aware access privilege [5].

6) DAC-MACS Framework

InDAC-MACS(Data Access Control For Multi-Authority Cloud Storage) framework a multi authority CP-ABE scheme is used where each attribute authority maintains disjoint attribute set which is proposed to provide efficient decryption and efficient attribute revocation method for it, which is then applied to get effective data access control with multiple attribute authorities in cloud storage system. This framework consists of five entities: a Certificate Authority(CA), the Attribute Authorities(AAs),the cloud server(server),the data owners(Owner),the data users(users) [10].

The CA is trusted Certificate Authority in the system who initializes the system and does registration of AAs and users. For every legal user CA assign a global unique user identity,legal user means user who has been authenticated with the system.CA also generates a pair of global secret key and global public key for this user. AA is an attribute authority who independently issues, revokes and updates user's attributes according to their identity in the domain. Every attribute is associated with single AA and each AA maintains arbitrary number of attributes. Each AA generates public attribute key for each attribute it manages and secret key for the user who possesses same attributes. The cloud server stores the encrypted data of the owner and allows data access to legal users. It generates cipher text decryption token using secret keys generated by AAs for the users. Using that decryption token user can decrypt the cipher text.

The server can generate correct decryption token only when the attributes satisfy the access policy defined in the cipher text. To get cipher text decryption token user has to submit global public key and secret key generated by some AAs to the server .After server generates decryption token, using this token along with global secret key user can decrypt the cipher text. The server does cipher text update when attribute revocation happens. Every owner divides his/her data into several components depending on logical granularities and encrypts each component with content keys using symmetric key algorithms. These content keys are encrypted by the access policy defined by data owner over attributes from different attribute authorities. Then the owner sends the encrypted data along with the cipher textto the server. Thus it provides efficient decryption method using token based decryption.

It also provides efficient attribute revocation method in multi-authority CP-ABE scheme which facilitates both forward security and backward security. It is efficient means it occurs with less communication and less computation cost.The revoked users can't decrypt the new ciphertext as it needs revoked attributes to decrypt. This is called backward security. The new user can also decrypt the earlier published data encrypted with earlier public keys, if it has sufficient attributes. This is called forward security. Even though it provides efficient decryption method and efficient attribute revocation method it leads to single point bottleneck problem as multiple attribute authorities acts a single authority since each AA maintain disjoint attribute set.

7)TMACS Framework

In TMACS (Threshold Multi-Authority Access Control System) with multiple authorities for public cloud storage system, CP-ABE scheme is used to guarantee data owners to get direct control over their data. In previous multi authority schemes like DAC-MACS, multiple attribute authorities maintain disjoint attribute subsets; however it results in single point bottleneck problem. In order to solve this, TMACS is proposed where multiple authorities jointly manage a uniform attribute set.

In this framework master key is shared among multiple attribute authorities by taking advantage of (t,n) threshold secret sharing and any legal user is able to generate his/her secret key by interacting with any t attribute authorities among n authorities.

In this framework there exists five entities: a global Certificate Authority(CA), Attribute Authorities(AAs),Cloud Servers(servers),Data Owners(Owners),Data Consumers(Users).The CA is global trusted entity responsible for setting up the system and initializing system parameters. It is also responsible for assigning unique *uid* for legal user and unique *aid* for each AA. It also determines the threshold value't' for AAs who are involved in the secret key generation at each time. The AAs do attribute management and key generation. Unlike previous multi-authority CP-ABE scheme, AAs jointly maintain the whole attribute set. All AAs cooperate with each other for sharing master key among them that means each AA gets its share of mater key as it private key. Then each AA sends its public key to CA to generate one of system public keys. When it comes to the generation of secret keys, AA independently generates corresponding secret key and there is no communication between AAs in secret key generation phase. The data owner encrypts his/her data with symmetric key algorithm and symmetric key will be encrypted under the access policy defined by the owner over the attributes, according to the attribute public keys generated by CA.

Now the owner sends the cipher text along with the encrypted symmetric key to store in the cloud server. The legal user can get the interested cipher text from the server, however user can decrypt the cipher text if and only if the access policy formulated by owner satisfies, the attribute set possessed by the user. The cloud server provides the owner platform for storing and sharing their data. Security and Performance analysis shows that TMACS is secure when less than 't' attribute authorities are compromised and also robust when all the't' attribute authorities are alive in the system. This framework solves the problem of single point bottleneck on both security and performance; however it is not efficient because user has to interact with‘t’ authorities and thus adds higher interaction overhead.

III SYSTEM ARCHITECTURE

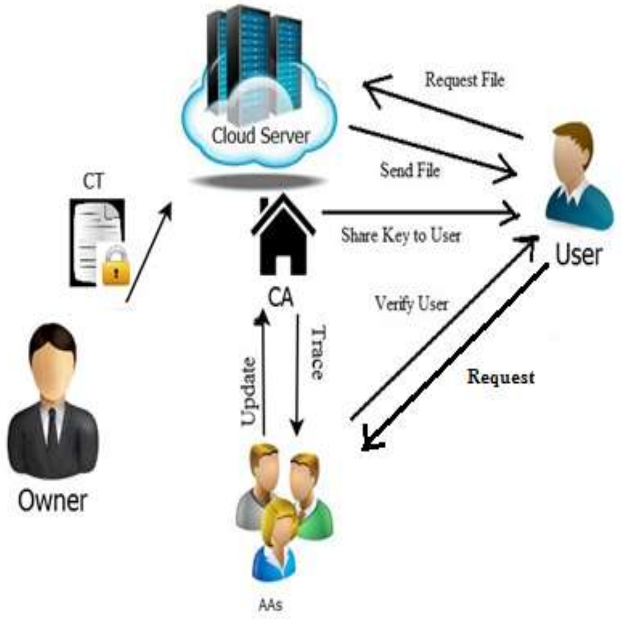

Figure1: System Architecture.

It proposes a novel heterogeneous framework to remove the problem of single-point performance bottleneck and provide a more efficient access control scheme with an auditing mechanism. This framework employs multiple attribute authorities to share the load of user legitimacy verification. Meanwhile, in this scheme, a central authority is introduced to generate secret keys for legitimacy verified users.

Methodology

1) Data Owner
Data owner encrypts the data with symmetric key algorithm. He/She defines the access policy over an attribute set and then encrypts the symmetric key under thepolicy according to the public keys obtained by CA. Data owner is verified for its legitimacy during registration and data is also verified before uploading.

2) User
The data user (consumer) is assigned a global user identity Uid by CA. It can get any interested encrypted data from the cloud and the user can decrypt the encrypted data if and only if its attribute set satisfy the access policy.

3) Central Authority (CA)
It is the administrator of the entire system. It helps in system construction by setting up system parameters and generating public key for attribute of universal attribute set. It generates unique ids for AAs and users after registration. It generates secrete keys for legitimacy verified users. It has capacity to trace which AA has maliciously verified a user.

4) Attribute Authorities (AAs)
The attribute authority (AAs) manages the whole attribute set individually so it can perform legitimacy verification of any user independently. AAs verify users legitimate attributes and generates intermediate key to assist CA to generate secret keys.

5) Cloud Server
Cloud servers provide public platform for data owners to store and share their encrypted data. Encrypted data can be freely downloaded by any user.

IV ADVANTAGES

✓ The Proposed System is efficient and scalable.
✓ Data confidentiality.
✓ Provides Data Security.

V CONCLUSION AND FUTURE WORK

It has been proposed a new heterogeneous framework to eliminate the single point performance bottleneck and increase the efficiency of the existing CP-ABE schemes. By effectively reformulating CP-ABE cryptographic technique into this novel framework, the proposed scheme provides a fine-grained, robust and efficient access control with one-CA/multi-AAs for public cloud storage. This scheme employs multiple AAs to share the load of the time-consuming legitimacy verification and standby for serving new arrivals of users requests. It has been proposed an auditing method to trace an attribute authority’s potential misbehavior. It has been conducted detailed security and performance analysis to verify that this scheme is secure and efficient. The security analysis shows that the scheme could effectively resist to individual and colluded malicious users, as well as the honest-but-curious cloud servers.

The system can be further improved by increasing the security, as it is mentioned that CA is assumed to be trustworthy, however we can check its behavior and take action if there is any discrepancy. This will surely make the system more secure and efficient.

REFERENCES

[1] Zhangjie Fu, Kui Ren, “Enabling personalized search over encrypted outsourced data with efficiency improvement” 2015 IEEE.

[2] Zhangjie Fu, Xingming Sun and Sai Ji, “Towards efficient content-aware search over encrypted outsourced data in cloud” IEEE INFOCOM 2016

[3] Kaiping Xue, “A dynamic secure group sharing framework in public cloud computing” 2013 IEEE.

[4] Attribute-based access to scalable media in cloud-assisted content sharing

[5] Junbeom Hur "Improving security and efficiency in attribute based data sharing" 2013 IEEE.

[6] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems,"IEEE Transactions on Parallel and Distributed Systems,vol. 22, no. 7, pp. 1214–1221, 2011.

[7] Jianan Hong, Kaiping Xue "TAFC: Time and attribute factors combined access control on time sensitive data in public cloud" 2015 IEEE

[8] Yingjie Xue, Jianan Hong, Wei Li and Kaiping Xue, "LABAC: A location-aware attribute-based access control scheme for cloud storage" 2016 IEEE

[9] K. Yang, X. Jia, K. Ren, and B. Zhang, "DAC-MACS:Effective data access control for multi-authority cloud storage systems" 2013 IEEE

[10] Jianwei Chen and Huadong Ma "Efficient decentralized attribute based access control for cloud storage with user revocation" 2014 IEEE

[11] Wei Li, Kaiping Xue, Yingjie Xue, and Jianan Hong "TMACS: A robust and verifiable threshold multi-authority access control system in public cloud storage" 2015 IEEE.