An Authentication Model to Access Encrypted Cloud Database Storage

U.Hari Prakash
Department of Computer Science Engineering
Sri Manakula Vinayagar Engineering College, Puducherry, India

Abstract— *Cloud computing provides highly scalable services for data storage over internet. In the cloud environment, where critical data is placed in infrastructures of untrusted third parties, ensuring data availability and security are very important. Database as a Service (DBS) is used to manage database in the cloud context. Secure DBS (Secure Database as a Service). Secure DBS supports multiple clients which are geographically distributed to execute the independent and concurrent operations on encrypted data in the untrusted cloud database storage. This architecture is designed to allow multiple and independent clients to connect directly to the cloud database storage without any intermediate servers. Secure DBS provides data confidentiality on both client and cloud level. The client can perform concurrent query processing on encrypted databases. Advanced Encryption Standard (AES) is a cryptographic technique used in the system to convert plain text to encrypted data. The encrypted data is stored on the untrusted cloud database storage with the guarantee of data confidentiality. The Secure DBS framework is enhanced to support concurrent database structure modification scheme with minimum overhead.*

Keywords— Secure DBS, AES, Cloud Computing

I. INTRODUCTION

Cloud computing is a recent trend in IT that moves computing and data away from desktop and portable PCs into large data centers. It refers to applications delivered as services over the Internet as well as to the actual cloud infrastructure namely, the hardware and systems software in data centers that provide these services. The key driving forces behind cloud computing is the ubiquity of broadband and wireless networking, falling storage costs, and progressive improvements in Internet computing software.

Cloud-service clients will be able to add more capacity at peak demand, reduce costs, experiment with new services and remove unneeded capacity, whereas service providers will increase utilization via multiplexing, and allow for larger investments in software and hardware. Consumers purchase such services in the form of infrastructure-as-a service (IaS), platform as a service (PaS), or software as a service (SaS) and sell value-added services to users. Within the cloud, the laws of probability give service providers great leverage through statistical multiplexing of varying workloads and easier management a single software installation can cover many users' needs.

II. RELATED WORKS

A. Secure DBS

Secure DBS is designed to allow multiple and independent clients to connect directly to the untrusted cloud DBS without any intermediate server. We assume that a tenant organization acquires a cloud database storage service from an untrusted provider. The tenant then deploys one or more machines and installs a Secure DBS client on each of them. This client allows a user to connect to the cloud DBS to administer it, to read and write data, and even to create and modify the database tables after creation.

We assume the same security model that is commonly adopted where tenant users are trusted, the network is untrusted and the cloud provider is honest-but-curious, cloud service operations are executed correctly, but tenant information confidentiality risk. The information managed by Secure DBS includes plaintext data, encrypted data, metadata, and encrypted metadata. Plaintext data consist of information that a tenant wants to store and process remotely in the cloud DBS.

B. Data Management

We assume that tenant data are saved in a relational database. We have to preserve the confidentiality of the stored data and even of the database structure because table and column names may yield information about saved data. We distinguish the strategies for encrypting the database structures and the tenant data.

Encrypted tenant data are stored through secure tables into the cloud database storage. To allow transparent execution of SQL statements, each plaintext table is transformed into a secure table because the cloud database storage is untrusted. The name of a secure table is generated by encrypting the name of the corresponding plaintext table. Table names are encrypted by means of the same encryption algorithm and an encryption key that is known to all the Secure DBS clients. Hence, the encrypted name can be computed from the plaintext name. On the other hand, column names of secure tables are randomly generated by Secure DBS. Hence, even if different plaintext tables have columns with the same name, the names of the columns of the corresponding secure tables are different. This design choice improves confidentiality by preventing adversarial cloud database storage from guessing relations among different secure tables through the identification of columns having the same encrypted name.

Secure DBS offers three field confidentiality attributes:

- Column (COL) default confidentiality level that should be used when SQL statements operate on one column; the values of this column are encrypted through a randomly generated encryption key that is not used by any other column.
- Multi column (MCOL) should be used for columns referenced by join operators, foreign keys, and other operations involving two columns; the two columns are encrypted through the same key.
- Database (DBC) recommended when operations involve multiple columns; in this instance, it is convenient to use the special encryption key that is generated and implicitly shared among all the columns of the database characterized by the same secure type.

C. Sequential SQL Operation

We describe the SQL operations in Secure DBS by considering an initial simple scenario in which we assume that the cloud database storage is accessed by one client. Our goal here is to highlight the main processing steps; we do not take into account performance optimizations and concurrency. The first connection of the client with the cloud DBS is for authentication purposes.

D. Concurrent SQL Operation

The support to concurrent execution of SQL statements issued by multiple independent clients is one of the most important benefits of Secure DBS with respect to state-of-the-art solutions. Our architecture must guarantee consistency among encrypted tenant data and encrypted metadata because corrupted or out-of-date metadata would prevent clients from decoding encrypted tenant data resulting in permanent data losses. A thorough integrity verification mechanism is integrated with the system.

Encrypted query submission model is used to secure the query values. Access control mechanism is used to allow users to grant permissions for other users. Cloud database storage security scheme is enhanced with data verification mechanism.

III. PROPOSED SYSTEM

We propose an architecture that integrates cloud database storage services with data confidentiality and the possibility of ensuring it by encrypting data. Secure DBS is designed to allow multiple and independent clients to connect directly to the untrusted cloud DBS without any intermediate server and to execute concurrent and independent operations on encrypted data.

The proposed architecture has the further advantage of eliminating intermediate proxies that limit the elasticity, availability, and scalability properties that are intrinsic in cloud-based solutions. Data, data structures and metadata are encrypted before upload to the cloud.

Multiple cryptography techniques are used to convert plain text into encrypted data. Table names and their column names are also encrypted in the cloud database storage security scheme. It includes SQL statements that modify the database structure. It provides consistency at the client and cloud level. It does not require a trusted broker or a trusted proxy because tenant data and metadata stored by the cloud database storage are always encrypted.

IV. SYSTEM ARCHITECTURE

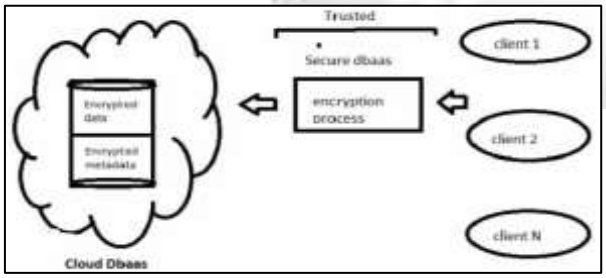

Fig. 1: System Architecture

Meta data will be stored after each encryption process and this metadata will become the index of the client's data in the cloud database storage. When client wants to retrieve the data from the cloud it requests a metadata and this metadata inquires in the cloud database storage for the data and the decryption is carried to plain text. Thus a client can retrieve his data from the data base.

V. 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.

A. Modules

The system consists of three modules namely,

- Tenant Admin
- User
- Tenant
- Secure DBS

B. Module Description

1) Tenant Admin

Multi-Tenancy concept can be achieved through this module. Many tenants can access the applications and computing resources in the same cloud server. Under these circumstances, it is important to separate application and data for conflicts avoidance to enhance the system and data security. New Tenants are registered and their details are collected and stored in the cloud database storage. Individual tenant id is generated which enables the cloud server to isolate the data tenant wise. Tenant details are available to the cloud server as plaintext data.

The registered tenant details can be viewed by tenant admin. A new URL is generated for each tenant. In the admin module we have various tables like add vehicle details, vehicle types, vehicle tariffs, add and update customers, add accident details and add service details.

2) User

User can either register as a new client or login to the old account. Clients play a major role in the project. Clients can retrieve the necessary tenant data from the untrusted database through SQL statements that operate on the encrypted data.

The encrypted data is decrypted and made available to the client. In this domain the user can register or login if he has already registered for this service, if the user is using the services for the first time then the signup details such as name, address, mobile number, and the necessary user name and password for the user can be given by the user himself. After the registration is done the necessary details will be stored in the admin's database and now the user can book the necessary cab for the desired location.

3) Tenant

Tenant can login and upload the data on the cloud. View secure data; add Secured data and Update/Delete Secured data. These tenant data are encrypted and stored on the cloud server database. Tenants can choose the tables that

they want to keep confidential. These table data is encrypted before storing in the cloud database storage.

The tenant portal has a table of contents listed under the Tenant Registration column, in this domain the tenant that is the user can give all his personal details for the registration purposes after entering certain necessary details and when the submit button is clicked all the entered details will be saved to the admin's database.

4) Secure DBS

Metadata information are collected for each tenant and stored in the cloud server database. The table Metadata contains information such as table encryption key and table plaintext name in the database. Tenant wise, the metadata information is separated and maintained on the cloud server database.

The tenant can provide the encryption key for each table which is used for encryption purposes. Multiple cryptography techniques are used to convert plain text into encrypted data.

VI. CONCLUSION

Cloud database storage services are integrated with data confidentiality and concurrent access models. Secure database as a service (Secure DBS). Framework is used to manage data access in encrypted cloud database storages. The Secure DBS scheme is enhanced with data integrity features. Concurrent database structure modification and query security tasks are improved with security methods. The system eliminates the intermediate proxies in database management process.

Database structure modification mechanism is adopted for multi user environment. The system improves the availability and scalability features. The response time in query processing is reduced by the system.

REFERENCES

[1] Luca Ferretti, Michele Colajanni, and Mirco Marchetti "Distributed, Concurrent, and Independent Access to Encrypted Cloud database storages", IEEE Transaction on Parallel and Distributed System, Vol 25, No 2, Feb 2014,pp 437-446.

[2] A.J. Feldman, W.P. Zeller, M.J. Freedman, and E.W. Felten, "SPORC: Group Collaboration Using Untrusted Cloud Re- sources," Proc. Ninth USENIX Conf. Operating Systems Design and Implementation, Oct. 2010.

[3] J. Li, M. Krohn, D. Mazie`res, and D. Shasha, "Secure Untrusted Data Repository (SUNDR)," Proc. Sixth USENIX Conf. Opearting Systems Design and Implementation, Oct. 2004.

[4] V. Ganapathy, D. Thomas, T. Feder, H. Garcia-Molina, and R. Motwani, "Distributing Data for Secure Database Services," Proc. Fourth ACM Int'l Workshop Privacy andAnonymity in the Information Soc., Mar. 2011.

[5] R.A. Popa, C.M.S. Redfield, N. Zeldovich, and H. Balakrishnan, "CryptDB: Protecting Confidentiality with Encrypted Query Processing," Proc. 23rd ACM Symp. Operating Systems Principles, Oct. 2011.