ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 42 – 46 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp09.pdf

Effective Way of access Information from Encrypted Databases in Cloud Infrastructure

K. Kiran Kumar [1], R. Suresh [2]
[1] M. Tech Student, [2] Associate professor, Dept. of CSE, CREC, Tirupathi.
[1] kirankumar5808@gmail.com,
[2] creccsehod@gmail.com,

Abstract:-

Cloud platform which shares the virtual database instances to the requested users seated in different geographical locations to access the stored as well as shared information from the cloud. But clients are afraid of keeping the sensitive information in the hands of cloud provider because they have their own doubts that how they serve the stored information securely in the presence of un-trusted users. By the detail analysis, we came to know that there are several approaches to store the information securely in the cloud database. But the present approaches are lagging to provide confidentiality in certain situation which arises while processing the stored information from the database. We introduce a new plug-in for Database as a service that protects the stored data privacy and it allows users to perform concurrent operation on encrypted data anywhere from the world. Along with these it provides flexibility to change the database structure also.

I Introduction:

Cloud computing is a new computing paradigm that is engineered on virtualization, parallel and distributed computing, utility computing, and service-oriented design. Within the last many years, cloud computing has emerged mutually of the foremost potent paradigms within the IT business, Cloud computing may be a thought that treats the resources on the web as a unified entity, a cloud. Users simply use services while not worrying regarding however computation is completed and storage is managed. It focuses on coming up with cloud storage for hardiness, confidentiality, and functionality. The cloud storage system is taken into account as an outsized scale distributed storage system that consists of the many freelance storage servers. Knowledge hardiness may be a major demand for storage systems. a method to produce knowledge hardiness is to duplicate a message specified every storage server stores a replica of the message. A Cloud direction system (CDBMS) may be distributed information that delivers computing as a service rather than a product. It's the sharing of resources, software, and knowledge between multiply devices over a network that is generally the web. It's expected that this range can grow considerably within the future. Associate example of this is computer code as a Service, or SaaS, that is associate application that's delivered through the browser to customers. Cloud applications connect with a information that's being run on the cloud and have variable degrees of potency. Some square measure manually designed, some square measure preconfigured, and sa few square measure native. Native cloud databases square measure historically higher equipped and additional stable that those who square measure changed to adapt to the cloud. Cloud Computing has been visualized because the next-generation design of IT Enterprise. In cloud computing application computer code and knowledge bases square measure moving to the centralized massive data centers. This mechanism brings regarding several new challenges, that

ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 42 – 46 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp09.pdf

haven't been well understood. Security and privacy considerations, however, square measure among the highest considerations standing within the method of wider adoption of cloud. In cloud computing the most concern is to produce the safety to finish user to safeguard files or knowledge from unauthorized user. Security is that the main intention of any technology through that unauthorized trespasser cannot access your file or knowledge in cloud. we've got styled one planned design and design which will facilitate to write and rewrite the file at the user facet that give security to knowledge at rest yet as whereas moving. Cloud computing is currently days rising field as a result of its performance, high accessibility, low cost. Within the cloud several services square measure provided to the shopper by cloud. Knowledge store is main future that cloud service provides to the businesses to store immense quantity of storage capability. However still several firms don't seem to be able to implement cloud computing technology attributable to lack of correct security management policy and weakness in protection that cause several challenge in cloud computing. Cloud computing is web primarily based computing wherever virtual shared servers give computer code, infrastructure, platform, devices and different resources and hosting to computers on a pay-as-you-use basis. Users will access these services offered on the "internet cloud" while not having any previous information on managing the resources concerned. Thus, users will concentrate additional on the core business processes instead of outlay time on gaining information on resources required to manage their business processes. Attributable to its low value, robustness, flexibility and omnipresent nature, cloud computing is ever-changing the method entities manage their knowledge. However, various privacy concerns arise whenever potentially sensitive data is outsourced to the cloud. The planned theme prevents the cloud server from learning any probably sensitive plaintext within the outsourced databases. It also allows the database owner to delegate users to conducting content level fine-grained private search and decryption. Moreover, our theme supports non-public questioning whereby neither the information owner nor the cloud server learns query details.

II System Overview

The system mainly focuses on following

- Cloud database
- Metadata Management
- Encryption algorithm

Cloud database: 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.

Metadata Management: Metadata generated by SecureDBaaS contain all the information that is necessary to manage SQL statements over the encrypted database in a way transparent to the user. Metadata management strategies represent an original idea because SecureDBaaS is the first architecture storing all metadata in the untrusted cloud database together with the encrypted tenant data.

Encryption algorithm: Choosing the encryption algorithms used to encrypt and decrypt all the data stored in the database table.

ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 42 – 46 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp09.pdf

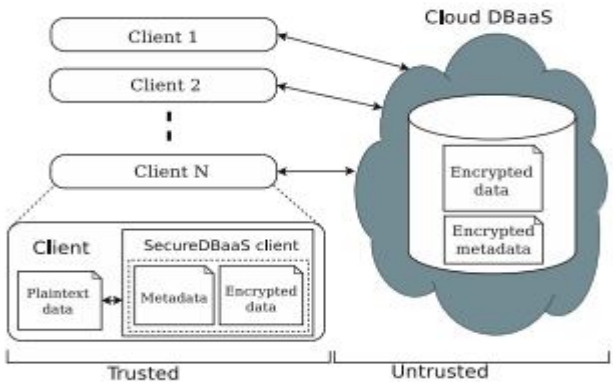

Fig.1: SecureDBaaS Architecture

Fig. 1 describes the overall architecture. We assume that a tenant organization acquires a cloud database service from an un-trusted DBaaS provider. The tenant then deploys one or more machines (Client 1 through N) and installs a SecureDBaaS client on each of them. This client allows a user to connect to the cloud DBaaS to administer it, to read and write data, and even to create and modify the database tables after creation. SecureDBaaS is designed to allow multiple and independent clients to connect directly to the untrusted cloud DBaaS without any intermediate server.

III System Design

3.1 Cloud database: 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.

3.2 Metadata Management: Metadata generated by SecureDBaaS contain all the information that is necessary to manage SQL statements over the encrypted database in a way transparent to the user. Metadata management strategies represent an original idea because SecureDBaaS is the first architecture storing all metadata in the un-trusted cloud database together with the encrypted tenant data.

3.3 Encryption algorithm: Choosing the encryption algorithms used to encrypt and decrypt all the data stored in the database table.

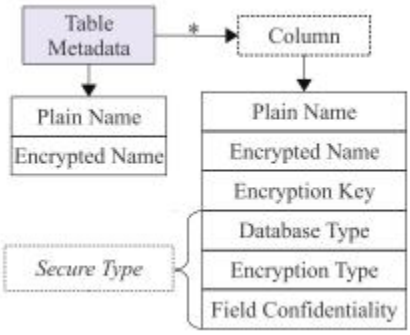

Fig. 2: Structure of table metadata.

IV IMPLEMENTATION

4.1 Data Management:

Cloud database acts as service provider for tenants. The cloud is created first for the system. All information or data store in the relational database. So for creating tables and column we have to access it with SQL query only.

4.2 Metadata Management:

Metadata generated by SecureDBaaS contain all the information that is necessary to manage SQL statements over the encrypted database in a way transparent to the user. Metadata management strategies represent an original idea because SecureDBaaS is the first architecture storing all metadata in the untrusted cloud database together with the encrypted tenant data.

SecureDBaaS uses two types of metadata.

- Database metadata are related to the whole database. There is only

One instance of this metadata type for each database.

- Table metadata are associated with one secure table. Each table metadata contains all information that is necessary to encrypt and decrypt data of the associated secure table.

ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 42 – 46 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp09.pdf

This design choice makes it possible to identify which metadata type is required to execute any SQL statement so that a SecureDBaaS client needs to fetch only the metadata related to the secure table/s that is/are involved in the SQL statement.

This design choice minimizes the amount of metadata that each SecureDBaaS client has to fetch from the un-trusted cloud database, thus reducing bandwidth consumption and processing time. Moreover, it allows multiple clients to access independently metadata related to different secure tables. Database metadata contain the encryption keys that are used for the secure types. A different encryption key is associated with all the possible combinations of data type and encryption type. Hence, the database metadata represent a key ring and do not contain any information about tenant data.

The structure of a table metadata is represented in Fig. 3. Table metadata contain the name of the related secure table and the unencrypted name of the related plaintext table. Moreover, table metadata include column metadata for each column of the related secure table. Each column metadata contain the following information.

- Plain name: the name of the corresponding column of the plaintext table.
- Coded name: the name of the column of the secure table. This is the only information that links a column to the corresponding plaintext column because column names of secure tables are randomly generated.
- Secure type: the secure type of the column. This allows a SecureDBaaS client to be informed about the data type and the encryption policies associated with a column.
- Encryption key: the key used to encrypt and decrypt all the data stored in the column.

SecureDBaaS stores metadata in the metadata storage table that is located in the untrusted cloud as the database. This is an original choice that augmnts flexibility, but opens two novel issues in terms of efficient data retrieval and data confidentiality. To allow SecureDBaaS clients to manipulate metadata through SQL statements, we save database and table metadata in a tabular form. Even metadata confidentiality is guaranteed through encryption. The structure of the metadata storage table is shown in Fig. 4 This table uses one row for the database metadata, and one row for each table metadata.

Database and table metadata are encrypted through the same encryption key before being saved. This encryption key is called a master key. Only trusted clients that already know the master key can decrypt the metadata and acquire information that is necessary to encrypt and decrypt tenant data. Each metadata can be retrieved by clients through an associated ID, which is the primary key of the metadata storage table. This ID is computed by applying a Message Authentication Code (MAC) function to the name of the object (database or table) described by the corresponding row. The use of a deterministic MAC function allows clients to retrieve the metadata of a given table by knowing its plaintext name. This mechanism has the further benefit of allowing clients to access each metadata independently, which is an important feature in concurrent environments. In addition, SecureDBaaS clients can use caching policies to reduce the bandwidth overhead.

Metadata Storage Table

ID	*Encrypted Metadata*	*Control Structure*
MAC('.'+Db)	Enc(Db metadata)	MAC(Db metadata)
MAC(T1)	Enc(T1 metadata)	MAC(T1 metadata)
MAC(T2)	Enc(T2 metadata)	MAC(T2 metadata)

Fig. 3: Organization of database metadata and table metadata in the metadata storage table.

ISSN 2347 - 3983
International Journal of Emerging Trends in Engineering Research (IJETER), Vol. 3 No.6, Pages : 42 – 46 (2015)
Special Issue of NCTET 2K15 - Held on June 13, 2015 in SV College of Engineering, Tirupati
http://warse.org/IJETER/static/pdf/Issue/NCTET2015sp09.pdf

4.3 Algorithms:

Encryption algorithms are applied to encrypt the database. There are various encryption algorithms symmetric and asymmetric, but we will apply symmetric algorithm which proved key distribution only once to all tenants there will be no different private key related to every user.

Conclusions:

In this paper, we have discussed concurrent and independent access to encrypted cloud databases, proposes an innovative architecture that guarantees confidentiality of data stored in public cloud databases. The proposed system will not require modifications to the cloud database, and it will be immediately applicable to existing cloud DBaaS. Resolve problem of single point failure and a bottleneck limiting availability and scalability of cloud database services.

REFERENCES

[1] M. Armbrust et al., “A View of Cloud Computing,” Comm. of the ACM, vol. 53, no. 4, pp. 50-58, 2010.

[2] W. Jansen and T. Grance, “Guidelines on Security and Privacy in Public Cloud Computing,” Technical Report Special Publication 800-144, NIST, 2011.

[3] A.J. Feldman, W.P. Zeller, M.J. Freedman, and E.W. Felten, “SPORC: Group Collaboration Using Untrusted Cloud Resources,” Proc. Ninth USENIX Conf. Operating Systems Design and Implementation, Oct. 2010.

[4] 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.

[5] P. Mahajan, S. Setty, S. Lee, A. Clement, L. Alvisi, M. Dahlin, and M. Walfish, “Depot: Cloud Storage with Minimal Trust,” ACM Trans. Computer Systems, vol. 29, no. 4, article 12, 2011.

[6] H. Hacigu ¨ mu ¨s¸, B. Iyer, and S. Mehrotra, “Providing Database as a Service,” Proc. 18th IEEE Int’l Conf. Data Eng., Feb. 2002.

[7] C. Gentry, “Fully Homomorphic Encryption Using Ideal Lattices,” Proc. 41st Ann. ACM Symp. Theory of Computing, May 2009.

[8] 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.