[Satve,3(7): July 2016]
DOI-10.5281/zenodo.57946

ISSN 2348 – 8034
Impact Factor- 4.022

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

INDEPENDENT ACCESS TO ENCRYPTED CLOUD DATABASES

Priyanka P.Satve[*1], Prof.H.A.Akarte[2] and Paresh P.Satve[3]

[*1]M.tech Student, Computer Department, Dr. Babasaheb Ambedkar Technological University, Lonere, India -402103.

[2]Assistant Professor, Computer Department, Dr. Babasaheb Ambedkar Technological University,Lonere, India -402103.

[3]B.E.Student, Information Technology, G.M. Vedak Institute Of Technology,Tala,India-402103.

ABSTRACT

In today's environment every user wants to access their data at any time and at anywhere. In an organization they store their data only on their computers, if they want to access their data during roaming situation, it is not possible one to carry the data at every time, this is a difficult factor for an organization. Cloud computing can address this problem by providing data storage mechanism to access the data at anywhere. This is one of the storage device used to access their data at anywhere through networks which is called cloud provider. For this service user is worried about the security and privacy issue. For this issue survey shows various techniques for the security and privacy mechanism for the user data. There are many data storage techniques available, but we are trying to combine cloud database service along with data security and also can perform independent operations on encrypted data.

Keywords: *Cloud, security, confidentiality, SecureDBaaS, database..*

1. INTRODUCTION

In a cloud context, where critical information is placed in infrastructures of untrusted third parties, ensuring data confidentiality is of paramount importance. This requirement imposes clear data management, choices of original plain data must be accessible only by trusted parties, that do not include cloud providers, intermediaries, and Internet; in any untrusted context, data must be encrypted. Satisfying these goals (to allow multiple, independent, and geographically distributed clients)has different levels of complexity depending on the type of cloud service. There are several solutions ensuring confidentiality for the storage as a service paradigm, while guaranteeing confidentiality in the database as a service (DBaaS) paradigm is still an open research area.

In this context, we propose SecureDBaaS as the first solution that allows cloud tenants to take full advantage of DBaaS qualities, such as availability, reliability, and elastic scalability, without exposing unencrypted data to the cloud provider. Unlike SecureDBaaS, architectures relying on a trusted intermediate proxy, do not support the most typical cloud scenario where geographically dispersed clients can concurrently issue read/write operations and data structure modifications to a cloud database.

2. METHODOLOGY

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and these steps are necessary to put transaction data in to a usable form for processing and it can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focus on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such way so that it provides security and ease of use with retaining the privacy.

OBJECTIVES

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the

[Satve,3(7): July 2016] ISSN 2348 – 8034
DOI-10.5281/zenodo.57946 Impact Factor- 4.022

data entry to handle large volume of data. It also provides record viewing facilities.When the data is entered it will check for its validity. Data can be entered with the help of screens.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined, how the information is to be displaced for immediate need and also for output in the form of hard copy. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making and also they should Identify the specific output that is needed to meet the requirements.

3. SYSTEM DESIGN

3.1 Data Owner

In this module, the data owner uploads their data in the cloud server. For the security purpose the data owner encrypts the data file and splits into four packets then store in the cloud. Data owner sends plain data to secure DBA(Trustee). The Data owner can have capable of manipulating the encrypted data file. And the data owner can set the access privilege to the encrypted data file.

3.2 Cloud Server

The cloud 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.

3.3 secure DBA(Trustee)

Secure DBA(Trustee) who is trusted to store verification parameters and offer public query services for these parameters. In our system, the secure DBA(Trustee) views the user data blocks and upload to the distributed cloud. In distributed cloud environment each cloud has user data blocks. If any modification tried by cloud owner a alert is send to the secure DBA(Trustee).

3.4 Data Consumer(End User / Group Member)

In this module, the user can only access the data file with the encrypted key, if the user has the privilege to access the file. For the user level, all the privileges are given by the data owner and the data users are controlled by the data owner only. Users can try to access data files either within their access privileges, so malicious users collude with each other to get sensitive files beyond their privileges.

3.5 Attacker (Unauthorized User)

Attacker adds the malicious data to a block in cloud server. Then the Unauthorized user will considered as a attacker.

[Satve,3(7): July 2016]
DOI-10.5281/zenodo.57946

ISSN 2348 – 8034
Impact Factor- 4.022

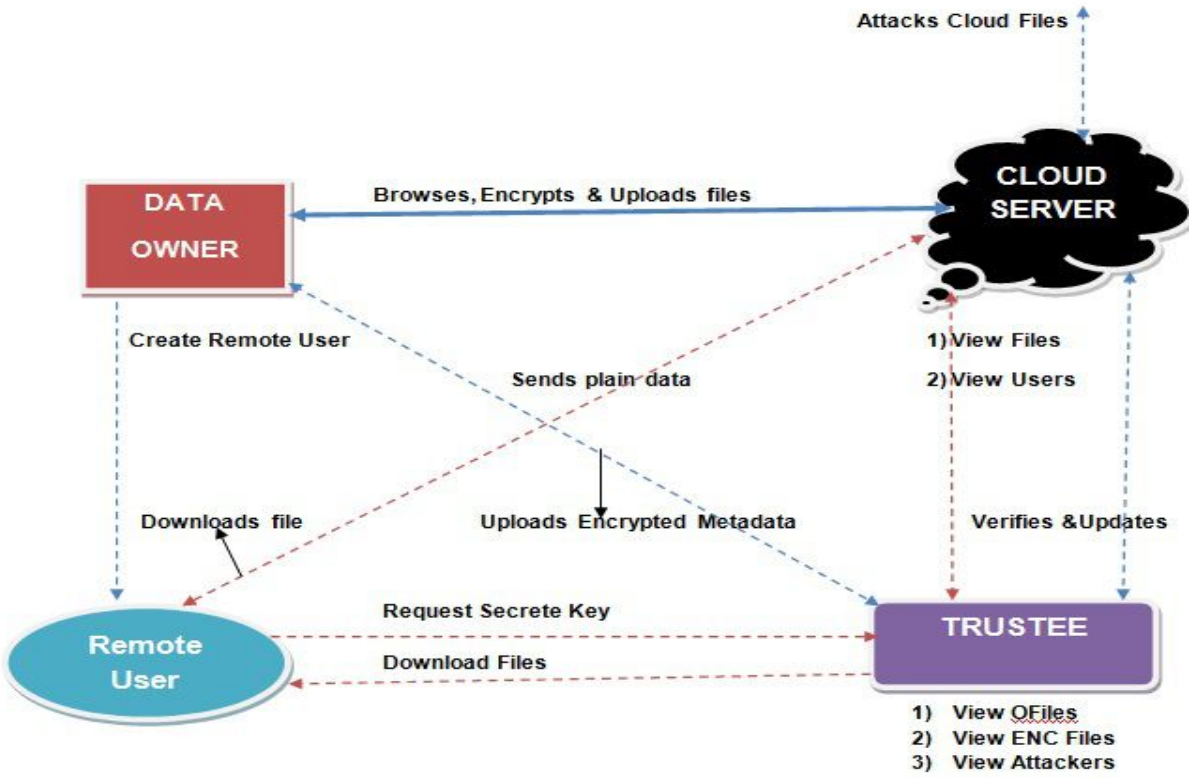

Figure 1 Architecture Diagram

4. MODULES

System modules and modules description are follows:

1. Setup Phase
2. Meta Data Module
3. Sequential SQL Operations
4. Concurrent SQL Operations

MODULES DESCRIPTION:

Setup Phase:

- We describe how to initialize Secure DBaaS architecture from a cloud database service acquired by a tenant from a cloud provider.
- We assume that the DBA creates the metadata storage table that at beginning contains just the database metadata, and not the table metadata. The DBA populates the database metadata through the Secure DBaaS client by using randomly generated encryption keys for any combinations of data types and encryption types, and store them in the metadata storage table after encryption through the master key.
- Then, the DBA distributes the master key to the legitimate users. User access control policies are administrated by the DBA through some standard data control language as in any unencrypted database. In the following steps, the DBA creates the tables of the encrypted database.

Meta Data Module:

- In this module, we develop Meta data. So our system does not require a trusted broker or a trusted proxy because tenant data and metadata stored by the cloud database are always encrypted.
- In this module, we design such as Tenant data, data structures, and metadata that must be encrypted before exiting from the client.The information managed by SecureDBaaS includes plaintext data, encrypted data,

[Satve,3(7): July 2016] ISSN 2348 – 8034
DOI-10.5281/zenodo.57946 Impact Factor- 4.022

metadata, and encrypted metadata. Plaintext data consist of information that a tenant wants to store and process remotely in the cloud DBaaS.

- SecureDBaaS clients also produce a set of metadata consisting of information required to encrypt and decrypt data as well as other administration information. Even metadata are encrypted and stored in the cloud DBaaS.

Sequential SQL Operations:

- The first connection of the client with the cloud DBaaS is for authentication purposes. Secure DBaaS relies on standard authentication and authorization mechanisms pro-vided by the original DBMS server. After the authentication, a user interacts with the cloud database through the Secure DBaaS client.
- Secure DBaaS analyzes the original operation to identify which tables are involved and to retrieve their metadata from the cloud database. The metadata are decrypted through the master key and their information is used to translate the original plain SQL into a query that operates on the encrypted database.
- Translated operations contain neither plaintext database (table and column names) nor plaintext tenant data. Nevertheless, they are valid SQL operations that Secure DBaaS client can issue to the cloud database. Translated operations are then executed by the cloud database over the encrypted tenant data. As there is a one-to-one correspondence between plaintext tables and encrypted tables, it is possible to prevent a trusted database user from accessing or modifying some tenant data by granting limited privileges on some tables.
- User privileges can be managed directly by the untrusted and encrypted cloud database. The results of the translated query that includes encrypted tenant data and metadata are received by the Secure DBaaS client; it is decrypted, and delivered to the user. The complexity of the translation process depends on the type of SQL statement.

Concurrent SQL Operations:

- The support to concurrent execution of SQL statements issued by multiple independent (and possibly geographically distributed) clients is one of the most important benefits of Secure DBaaS 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.

5. DATA FLOW DIAGRAM

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to representa system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system. The dataflow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

ISSN 2348 – 8034
Impact Factor- 4.022

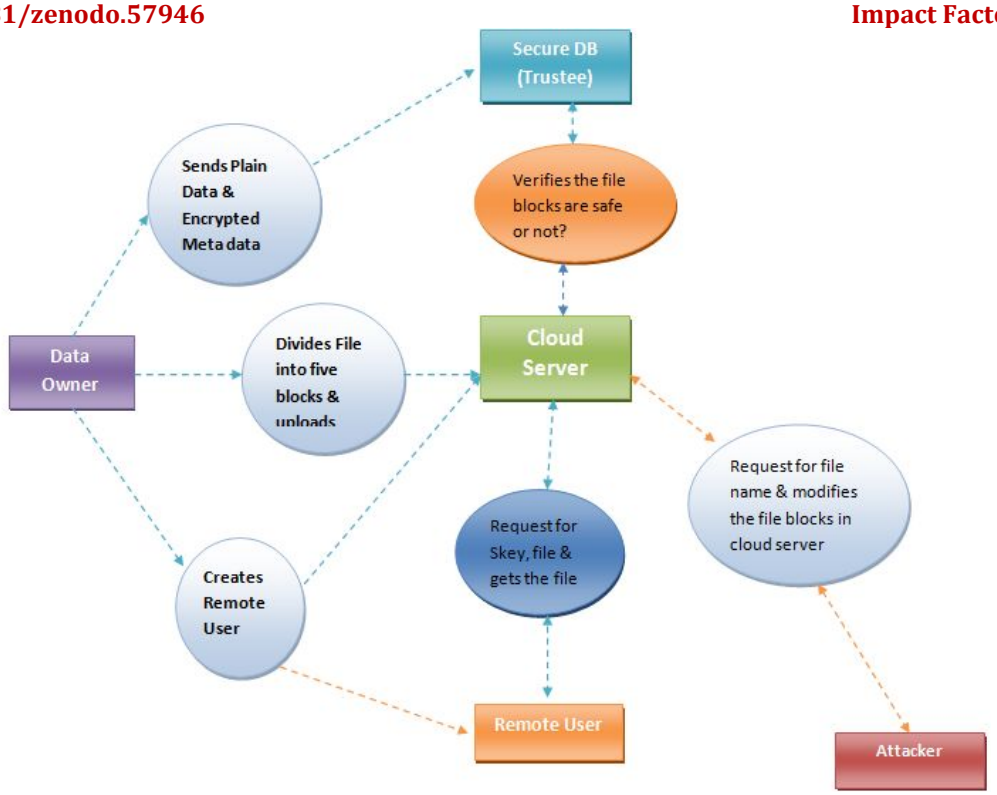

Figure 2 DFD Diagram

6. CONCLUSION

We propose an innovative architecture that guarantees confidentiality of data stored in public cloud databases. Unlike state-of-the-art approaches, our solution does not rely on an intermediate proxy that we consider a single point of failure and a bottleneck limiting availability and scalability of typical cloud database services. A large part of the research includes solutions to support concurrent SQL operations (including statements modifying the database structure) on encrypted data issued by heterogenous and possibly geographically dispersed clients. The proposed architecture does not require modifications to the cloud database, and it is immediately applicable to existing cloud DBaaS, such as the experimented PostgreSQL Plus Cloud Database, Windows Azure, and Xeround. There are no theoretical and practical limits to extend our solution to other platforms and to include new encryption algorithms. It is worth observing that experimental results based on the TPC-C standard benchmark show that the performance impact of data encryption on response time becomes negligible because it is masked by network latencies that are typical of cloud scenarios. In particular, concurrent read and write operations that do not modify the structure of the encrypted database cause negligible overhead. Dynamic scenarios characterized by (possibly) concurrent modifications of the database structure are supported, but at the price of high computational costs. These performance results open the space to future improvements that we are investigating.

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] H. Hacigu¨mu¨ s¸, B. Iyer, and S. Mehrotra,"Providing Database as a Service," Proc. 18th IEEE Int'l Conf. Data Eng., Feb. 2002.
[4] C. Gentry, "Fully Homomorphic Encryption Using Ideal Lattices," Proc. 41st Ann. ACM Symp. Theory of Computing, May 2009.
[5] D. Agrawal, A.E. Abbadi, F. Emekci, and A. Metwally, "Database Management as a Service: Challenges and Opportunities," Proc. 25th IEEE Int'l Conf. Data Eng., Mar.-Apr. 2009.

[Satve,3(7): July 2016] ISSN 2348 – 8034
DOI-10.5281/zenodo.57946 Impact Factor- 4.022
[6] G. Cattaneo, L. Catuogno, A.D. Sorbo, and P. Persiano, "The Design and Implementation of a Transparent Cryptographic File System For Unix," Proc. FREENIX Track: 2001 USENIX Ann. Technical Conf., Apr. 2001.
[7] L. Ferretti, M. Colajanni, and M. Marchetti,"Supporting Security and Consistency for Cloud Database," Proc. Fourth Int'l Symp. Cyberspace Safety and Security, Dec. 2012.