ISSN: 2321-7782 (Online)
Volume 3, Issue 11, November 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Encrypting Cloud Databases for Security by Using Secure DBaaS

Soshanka Khumanthem[1]
Computer Engineering (Computer Networks)
College of Engineering & Management Research
Pune - India

Nagaraju Bogiri[2]
Professor
Computer Engineering (Computer Networks)
College of Engineering & Management Research
Pune - India

Abstract: With the increasing growth in communication, storage and processing information allow us to place all data into some remote place where every data can be managed. Cloud environment becomes very exciting place for the readying of huge scale applications attributable to their extremely ascendible and offered infrastructure. Information as a Service (DBaaS) model is used to manage databases in cloud setting. The user can access and extract the sensitive information whenever they want from cloud data storage capacity. Here storing and accessing of data is done through third party server or proxy server. This increases the load of the user's memory and time complexity. And also this leads to the arise of security issue. There are several alternatives exist for storage services , but still data confidentiality solutions for the database as a service paradigm are still immature. In this paper we propose a novel architecture that implements cloud database services with confidentiality of data and the opportunity of performing parallel operations on encrypted data. This architecture is the first solution that supports physically scattered clients which are directly connected to an encrypted cloud database, and to perform concurrent and independent operations such read, write and modify the database structure. Further advantages of this proposed architecture is that it eliminates intermediate proxy servers that limit the availability, elasticity, and scalability properties cloud-based solutions.

Key words: cloud environment, DBaaS, SecureDBaaS, proxy server, security, confidentiality.

I. Introduction

In today's world information becomes more valuable in every one life. So they try to store information where it has portable. User needs to access that information whenever they need it. To make this happen, this information needs to be uploaded to some central data repositories through network. Some Organizations provide facility for fast and reliable access to information. Organizations usually maintain one or more data centres to store and manage information.

For storing a large amount of data into network, cloud computing comes into picture. Cloud computing helps the user in storing, retrieving and updating data present in the cloud. A cloud computing provides solution for data storage issue, as Database-as-a-Service. The data that are store in the cloud typically use primitive file storage system rather than databases. Database-as-a-Service(DBaaS) helps the user in accessing the required data through the internet.

Cloud computing is a tool that offers enormous benefits to its subscribers. Since it is a tool, it has two main issues. The major concern is security and privacy in cloud. In this cloud context, user critical information is placed in infrastructure of un-trusted third parties, ensuring data confidentiality is of great importance.

We required the original plain data to be accessible only by trusted parties that do not include cloud providers, intermediaries and internet, in any un-trusted context, data must be encrypted. There are several other solutions that ensure confidentiality for the storage as a service paradigm. Database as a service(DBaaS) is one of them but it's still open research.

In this paper we proposed an architecture called SecureDBaaS for the cloud database. This takes full advantage as DBaaS qualities such as scalability, availability and reliability without revealing data to cloud provider. The architecture design in such a way that numerous independent clients can perform the operations on the encrypted data by the SQL statements, where the user can modify or update the database structure.

The architecture has the property of executing the independent and parallel operations to the remote encrypted database from any geographically located clients. In this system, the intermediate proxy between the client and the cloud provider is eliminated. This architecture can accomplish the same elasticity, availability and reliability of the DBaaS in cloud.

II. Literature Survey

With the rapid growth of information we need to store our data in a safe place. This required an organization for storing and managing the data from design to end of existence. Today all information or data can be stored in the internet storage space that is cyberspace. Cyberspace are delivered and retained by the third party through the internet. Cloud storage space gives large storage area where users can easily access and is always available for use. It has three major attributes: accessing data through Web services APIs on a non-persistent network connection, highly available huge quantity of storage space, and pay as per use model. It supports rapid scalability. The main advantage of cloud storage is that user can be access its data anytime from anywhere.

The evolution of Cloud Storage based on traditional network storage and hosted storage.

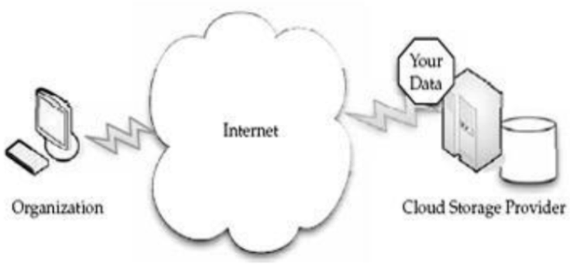

Fig1: simple cloud storage[6]

In cloud we can store data varying from small amount to entire warehouse of an organization. For utilizing cloud storage user has to pay to cloud storage provider. The payment is base upon the uses of cloud storage. In the cloud storage environment, the user data will be copied into on cloud data centre of the cloud. This data stays in data servers and made available on the cloud. This interaction between data centre will be result in high availability of the data server on cloud.

In cloud storage, cloud computing is done. Fig. 2 shows the evolution of Cloud Storage based on traditional network storage and hosted storage.

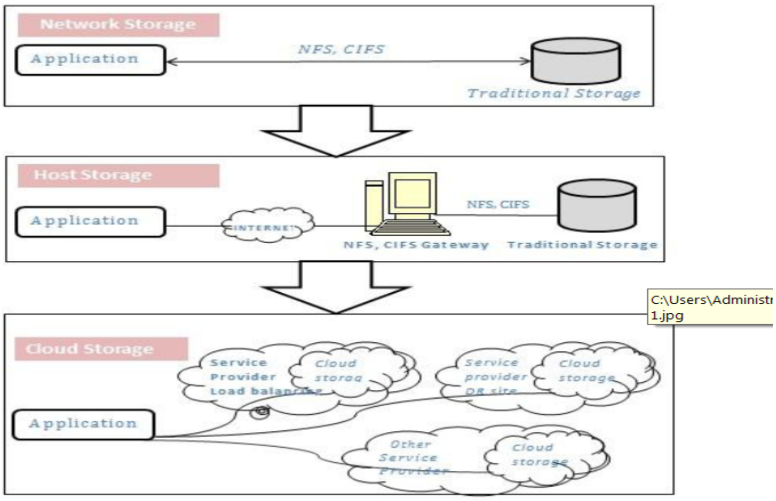

Fig 2: cloud storage evaluation[6]

Network storage: Network storage is a unit, where computer connected to a network that provides data storage to other devices on the network. It is only file-based data storage services. There is possible to run additional software on this network storage unit. It uses stripped down operating system frequently. Some protocol use in network storage are NFS, SMB/CIFS (Server Message Block/Common Internet File System), AFP (used with Apple Macintosh computers). It does not limit the clients to a single protocol.

Hosted storage: Hosted storage has a gateway between traditional storage and application. Cloud storage gateways use standard network protocols which integrate with existing applications. Some cloud gateways includes additional features like backup and recovery. It can also serves as an intermediaries to multiple cloud storage providers.

" A View of Cloud Computing" M. Armbrust [1], has shown that Cloud computing has the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even more attractive as a service and shaping the way IT hardware is designed and purchased. Developer's new idea for using internet services have outlays hardware to deploy their service or the human expense to operate it. Cloud computing will require much resources so developers much take it into account. Moreover:

1. Applications Software needs to both scale down rapidly as well as scale up, which is a new requirement. Such software are costly for licensing model to match needs of Cloud Computing.
2. Infrastructure Software is running on VMs. Moreover, billing needs to build in from the start. 3. Hardware Systems should be designed at the scale of a container (at least a dozen racks), which will be is the minimum purchase size.

"SPORC: Group Collaboration Using Un-trusted Cloud Resources" A.J. Feldman, W.P. Zeller, M.J. Freedman, and E.W. Felten [2], have describe, cloud services allow multiple users to edit shared state concurrently and in real-time, while being scalable, highly available, and globally accessible.. In SPORC, encrypted data is only examine by the server and cannot proceed to next step without being detected. SPORC allows concurrent, low-latency editing of shared state, permits disconnected operation, and supports dynamic access control even in the presence of concurrency Acknowledgments.

"Secure Un-trusted Data Repository (SUNDR)" J. Li, M. Krohn, D. Mazie` res, and D. Shasha, [3] have proposed a network file system called SUNDR where data can be securely store on untrusted servers.

SUNDR's has a property called fork consistency, it guarantees the client to detect any integrity or consistency failures happens as long as they see each other's file modifications. Measurements of our implementation show performance that is usually close to and sometimes better than the popular NFS file system.

"Depot: Cloud Storage with Minimal Trust" P. Mahajan, S. Setty, S. Lee, A. Clement, L. Alvisi, M. Dahlin, and M. Walfish [4] have described the design, implementation, and evaluation of Depot, a cloud storage system that minimizes trust assumptions. Depot began with an attempt to explore a radical point in the design space for cloud storage: trust no on

"Providing Database as a Service" H. Hacigu¨ mu¨ s, B. Iyer, and S. Mehrotra [5], have proposed a new concept regarding data management by a third party service provider hosts "database as a service". It provides customers to create, store, and access their databases at the host site. Here they introduced NetDB2, an internet-based database service built on top of DB2 that provides users with tools for application development, creating and loading tables, and performing queries and transactions.

III. Need

In cloud context, we are placing critical information to some un-trusted third parties, ensuring data confidentiality is of paramount importance. For the safe guard of our information, original plain data must be accessible only by trusted parties that do not include cloud providers, intermediaries, and internet; in any un-trusted context, data must be encrypted. In this paper we propose an architecture called SecureDBaaS where we can eliminate all intermediate server between the cloud client and the cloud provider. This architecture integrates cloud database services with data confidentiality and the possibility of executing concurrent operations on encrypted data. It also supports geographically located clients to connect directly to an encrypted database and execute concurrent and independent operations including modifying the database structure.

IV. Proposed System

Giving confidentiality to the data is much difficult in today's electronic world because every individuals, devices, and sensors are connected and information is created, accessed and shared widely with one another. Authentication methods are set up for this purpose. For storing data safely on the cloud, data must be encrypted before storing them. This increases the privacy of your data.

For storing data on the server cloud, the client must sign up first. Registration must be done strictly because login password is provided here. After signing up, the client can now login to his account. The information managed by SecureDBaaS includes plaintext data, encrypted data, metadata, and encrypted metadata. Plaintext data is the information that a tenant wants to store and process remotely in the cloud DBaaS. To prevent an un-trusted cloud provider from violating confidentiality of tenant data stored in plain form, SecureDBaaS adopts multiple cryptographic techniques to transform plaintext data into encrypted tenant data and encrypted tenant data structures because even the names of the tables and of their columns must be encrypted. SecureDBaaS clients produce also 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. In this architecture, tenant data is store in the cloud database, and save metadata in the client machine or split metadata between the cloud database and a trusted proxy. The client can retrieve data from the cloud server through SQL statements.

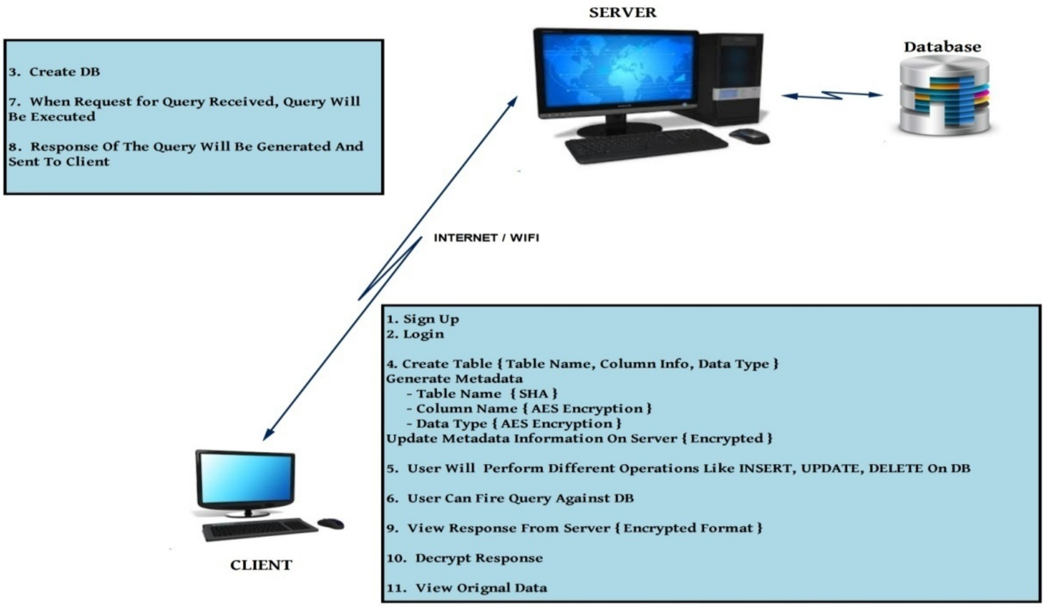
SERVER
Database
3. Create DB
7. When Request for Query Received, Query Will Be Executed
8. Response Of The Query Will Be Generated And Sent To Client
INTERNET / WIFI
1. Sign Up
2. Login
4. Create Table { Table Name, Column Info, Data Type }
Generate Metadata
- Table Name { SHA }
- Column Name { AES Encryption }
- Data Type { AES Encryption }
Update Metadata Information On Server { Encrypted }
5. User Will Perform Different Operations Like INSERT, UPDATE, DELETE On DB
6. User Can Fire Query Against DB
9. View Response From Server { Encrypted Format }
10. Decrypt Response
11. View Orignal Data
CLIENT

Fig.3 System Architecture

Metadata Storage Table:

Metadata is generated by SecureDBaaS and it includes all the information need to access the data from encrypted database. Metadata storage table is the table that stores metadata of SecureDbaaS and it is placed in cloud database. It is a flexible approach but come with two issues efficiency of data access and confidentiality.

To provide efficiency of data access SecureDBaaS use two metadata.

1. Database Metadata: This metadata associated to entire database. This metadata has a only one instance for each database in a cloud.
2. Table metadata: This is related with secure table. That is this meta-stable include all the information about encryption n decryption of secure table.

Same encryption key is used for encrypting database and table metadata before it has been stored at cloud database. This encryption key is known as master key. This key is known by trusted clients only. The used of this master key is to decrypt the metadata and obtain information that is needed to encrypt and decrypt data at cloud database. Each client has associated ID. Associated ID can retrieve client's metadata. For metadata storage table, this ID is the primary key.

Thus the clients are allowed to access metadata independently, which is a primary feature in concurrent environments. In addition, SecureDBaaS clients can use caching policies to reduce the bandwidth overhead.

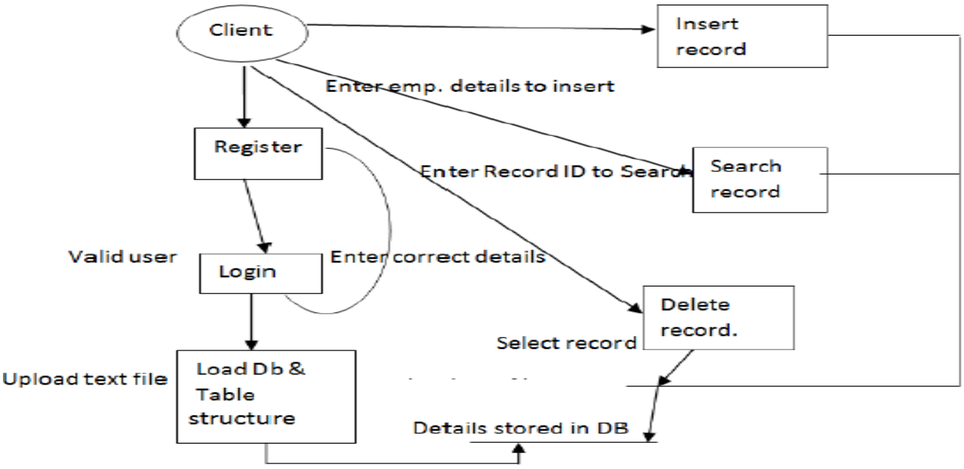

Fig.4 Data flow diagram

V. CONCLUSION

Giving confidentiality to the data is much difficult in today's electronic world as more data is being stored and exchanged. Since the technology becomes much advanced organizations are facing extremely complex risk matrix for assuring confidentiality for sensitive personal information. Security and confidentiality mechanism has become a more protective issue for those who are storing data in cloud. Encryption provides confidentiality for data stored in cloud; especially important when stored data is sensitive corporate data should not fall into the wrong hands. The SecureDBaaS architecture give confidentiality for data saved into cloud databases.

Those data which are stored on the cloud database are encrypted through cartographic algorithms and allows the execution of SQL queries on encrypted database. This architecture also provides multiple independent clients to access to data storing at cloud database. This architecture eliminates intermediate proxy that represents and also avoids the single point of failure and a system bottleneck, which in turn increases the availability and scalability of cloud database services.

ACKNOWLEDGMENT

I am grateful to my guide Prof. Nagaraju Bogiri , Department of Computer Science and Engineering, for being a constant guide and valuable suggestions and not only encouraging but helping me at every stage of my work.

References

1. M. Armbrust et al., "A View of Cloud Computing," Comm. of the ACM, vol. 53, no. 4, pp. 50-58, 2010.
2. 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.
3. J. Li, M. Krohn, D. Mazie` res, and D. Shasha, "Secure Untrusted Data Repository (SUNDR)," Proc. Sixth USENIX Conf. Operating Systems Design and Implementation, Oct. 2004.
4. P. Mahajan, S. Setty, S. Lee, A. Clement, L. Alvisi, M. Dahlin, and M. Walfish, "Depot: Cloud Storage with Minimal Trust", ACMTrans. Computer Systems, vol. 29, no. 4, article12, 2011.
5. H. Hacigu¨ mu¨ s¸, B. Iyer, and S. Mehrotra, "Providing Database as a Service", Proc. 18th IEEE Int'l Conf. Data Eng.Feb.2002.
6. Vijayalaxmi Joshi and Swathi Ptil, "SecureDBaaS Architecture For Encrypted Cloud Database," International Journal of Computer Application (2250-1797) Volume 5– No. 4, June2015.