International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Privacy Preserved Encrypted Cloud Data Access

Anes P. A.[1], Neethu Francis[2]

[1]M.Tech Scholar, KMP College of Engineering & Technology, Cherukunnam, Perumbavoor, India

[2]Asst Professor, Department of Computer Science & Engineering, KMP College of Engineering & Technology, Cherukunnam, India

Abstract: ***Several alternatives exist for storage services with the guarantee of security and availability for data at rest, in motion, and in use. In a cloud context, security can improve due to centralization of data, increased security-forced resources etc. The existing SecureDBaas architecture, integrates cloud database services with data confidentiality and the possibility of executing concurrent operations on encrypted data. This architecture has the further advantage of eliminating intermediate proxies. In this paper, propose architecture by providing security to SecureDBaaS. Here introducing a trusted third party to check whether any corruption occur at the cloud server database data.***

Keywords: Cloud Server Database – SecureDBaaS Architecture, Data confidentiality, Encryption, Trusted Third Party

1. Introduction

Cloud computing is the delivery of computing services over the Internet. The cloud computing model allows access to information and computer resources from anywhere that a network connection is available. The cloud computing service models are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). In a Software as a Service model, delivering application resources such as any required software, operating system, hardware, and network over Internet. In Platform as a Service (PaaS), an operating system, hardware, and network resources are provided as cloud service, and the customer installs or develops his own software and applications. The Infrastructure as a Service (IaaS) model provides just the hardware and network; the customer installs or develops its own operating systems, software and applications. Mainly there are three cloud database services: availability, elasticity and scalability. Availability means, the data should madeavailable to clients as their need. Elasticity and Scalability means that cloud computing offers unlimited processing and storage capacity. Security is one of the most often-cited objections to cloud computing. Security is often as good as or better than other traditional old systems, in part because providers are able to devote resources to solving security issues that many customers cannot afford to tackle. However, the complexity of data/information security is greatly increased when data is distributed over a wider area or over a greater number of devices or equipments, as well as, in multi-tenant systems shared by unrelated users. In a cloud context, where critical information is placed in infrastructures of untrusted third parties, ensuring data confidentiality is of prime importance [1], [2]. This requirement imposes clear data management choices: original plain data must be accessible only by trusted parties that do not include cloud service providers, intermediaries, and Internet; in any untrusted environment, data must be encrypted.

The existing SecureDBaaS architecture is the first solution that allows cloud tenants to take full advantage of DBaaS qualities, such as elastic availability, reliability, and scalability, without exposing unencrypted data to the cloud provider. Here combining the cloud database services with data confidentiality is demonstrated through a prototype of SecureDBaaS that supports the execution of concurrent and independent operations to the remote encrypted database from many geographically distributed clients as in any unencrypted DBaaS setup[3].

In this context, propose an architecture which combines security feature with existing SecureDBaaS architecture. The architecture design was motivated by a threefold goal: multiple clients can access concurrently and independently a cloud database service, preserve data confidentiality by allowing a cloud database server to execute concurrent operations over encrypted data, to eliminate a trusted broker or trusted proxy.

2. Review of Secure DBaaS Architecture

The SecureDBaaS architecture integrates cloud database services with data confidentiality and the possibility of executing concurrent operations on encrypted data. This is the first solution supporting distributed clients to connect directly to an encrypted cloud database system, and to execute concurrent and independent operations including those modifying the database structure [4]. The proposed architecture has the further advantage of eliminating intermediate proxies that limit the elastic scalability and availability properties that are intrinsic in cloud-based solutions.

The SecureDBaaS architecture make use of the cloud platforms and does not introduce any intermediary proxy or broker server between the client and the cloud service provider. Eliminating any trusted intermediate server or proxy allows SecureDBaaS to achieve the same elastic availability and reliability levels of a cloud DBaaS. Other proposals (e.g., [4], [5], [6], [7]) based on intermediate server(s) were considered impracticable for a cloud-based solution because any proxy represents a single point of failure and a system bottleneck that limits the main benefits (e.g., scalability, availability, and elasticity) of a database service deployed on a cloud environment. 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.

A proxy-based architecture requiring that any client operation should pass through one intermediate server is not suitable to cloud-based infrastructure, in which multiple clients, typically distributed among various locations, need concurrent access to data stored in the same DBMS. SecureDBaaS supports distributed clients issuing independent and concurrent SQL operations to the same database and possibly to the same data. In the cloud DBaaS, only the table data is stored. Before storing the data, it should be encrypted. The encrypted data is stored in the columns and rows of the table. So that the table name, column name and row name should be encrypted by using any encryption algorithm. The clients can access these encrypted table data by using SQL operations. The SQL operations are used to read and write data and even to make modifications to the database. The SQL operations are issued by either a single client or by multiple distributed clients. The SQL operations are used for concurrent access to encrypted data by geographically distributed clients.

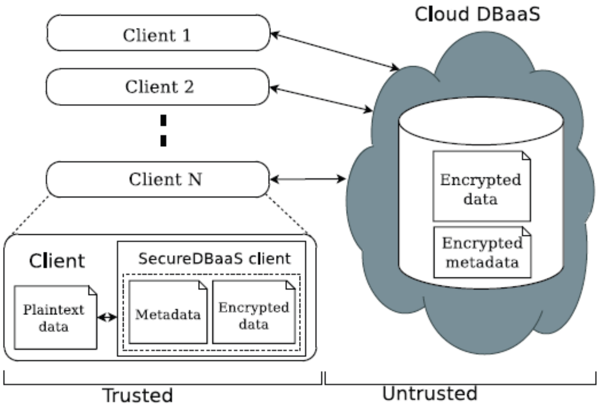

Figure 1: Existing SecureDBaaS architecture

Figure 1 shows the structure of existing system. Here the tenant organization deploys one or more clients. These clients are directly connected to the cloud DBaaS. In the cloud DBaaS, it stores only the table data in the encrypted format. So the clients can access the data through SQL operations. There is not a mechanism to check the correctness of encrypted data in the database. If there occurred any unauthorized modifications to the data in the cloud DBaaS, it will not inform to the owner of data.

Generally, the main two drawbacks of existing system are: it only manages table data in the cloud server database; there occur corruptions to the encrypted data at the database. The proposed system eliminates these two drawbacks and is also provide data confidentiality along with the cloud database services.

3. Proposed System

In the cloud computing environment, there occur a chance of corruption to the encrypted data at the cloud server database. This problem can be solved by introducing a trusted third party. The third party can periodically check whether any corruption occur at the database. The trusted party is connected only towards the cloud server database; there is no connection between the client and the trusted party.

In the existing SecureDBaaS architecture, only the table data is stored in the cloud server database. But in this context, whole types of data stored in the database. Main aim of this paper is to increase the security. In this, cloud is used as a database storing our data or files. For the security, before saving the file into the cloud database encrypting it by using an encryption algorithm. And only the authenticated user can access the encrypted file.

The encrypted data at the database may get corrupted by untrusted parties. The newly inserted trusted third party observe the database each time for checking whether any attempts at unauthorized file modification by malicious server operators or users. Also the proposed system provides whole services of SecureDBaaS architecture. Generally the main features of proposed system are.

- It guarantees data confidentiality for whole types of data by allowing a cloud database server to execute concurrent operations over encrypted data.
- Multiple clients can access concurrently and independently a cloud database service.
- It provides same availability, elasticity, and scalability of the original cloud DBaaS.
- It does not require a trusted intermediate proxy.
- Trusted third party detects unauthorized modifications to encrypted data at the database.

4. Encryption Schemes

The Homomorphic encryption scheme has special significance over encrypted data SQL operations. But we cannot apply fully Homomorphic encryption due to resource excessive operations. So can use RSA or AES algorithm for the purpose. RSA is simple to implement compared to AES.

5. Architecture Design

In this section mainly describe about the structure of proposed system. The architecture mainly consist of a cloud database server, number of clients and a trusted third party. SecureDBaaS is designed to allow multiple and independent clients to connect directly to the untrusted cloudDBaaS without any intermediate server or proxy. A trusted third party is connected towards the cloud DBaaS. Fig. 2 depicts the overall architecture.

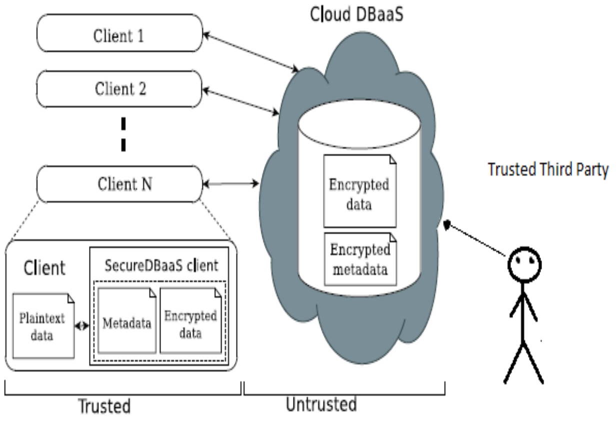

Figure 2: SecureDBaaS architecture with security

Assume that a tenant organization acquires a cloud database service from an untrusted DBaaS provider. The tenant organization or department then deploys one or more machines (Client 1 through N) and installs a *SecureDBaaS client* on each of them. This client console allows a user to connect to the cloud DBaaS to administer it and to read and write data. The existing a trusted third party, which is connected to the cloud server database. The tenant data must be encrypted before exiting from the client.

The information managed by SecureDBaaS includes *plaintext tenant data, encrypted tenant data, metadata, and encrypted metadata. Plaintext tenant data* consist of information that a tenant organization or department wants to store and process remotely in the cloud DBaaS. To prevent an untrusted cloud provider from violating confidentiality of tenant data stored in plain form, SecureDBaaS adopts cryptographic techniques to transform plaintext tenant 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 (users) produce also a set of metadata constitute information required to encrypt and decrypt data as well as other administration information.

SecureDBaaS moves away from existing architectures that store just tenant data in the cloud database, and save metadata in the client machine [5] or split metadata between the cloud database and a trusted proxy [4]. When dealing with scenarios where multiple clients can access the same database concurrently, previous solutions are inefficient. SecureDBaaS proposes a different approach where all data and metadata are stored in the cloud database. SecureDBaaS clients can retrieve the necessary metadata from the untrusted database, so that multiple instances of the SecureDBaaS client can access to the untrusted cloud database independently with the guarantee of the equivalent availability and scalability properties of typical cloud DBaaS.

In the proposed system additionally a trusted third party is inserted. This third party is used to check the correctness of data in the cloud database system. In the cloud database the data is stored in encrypted format to avoid the unauthorized file modification by malicious server operators or users. There occurs a chance for the corruption to the encrypted data. The newly inserted third party periodically check the encrypted data in the database and announces error whether any corruption occur in it.

In the previous work only the table data is stored in the database. But in the proposed system, whole type of data is stored. The data must be encrypted before storing into the database by using any type of encryption strategies. These additional features make it as an efficient storage management system.

6. Conclusion

Cloud computing is the delivery of computing services over the Internet technology. Without knowing, many people use cloud computing services for their own personal needs. In a cloud context, where critical information is placed in infrastructures of untrusted third parties, ensuring data confidentiality is of prime importance. Proposed system provides an innovative architecture that guarantees confidentiality of data stored in public cloud databases. The proposed system combines the security feature with existing SecureDBaaS architecture by introducing a trusted third party. This trusted third party checks whether any corruption occur to encrypted data at the cloud server database. This solution does not rely on an intermediate proxy and it also includes solutions to support concurrent operations on encrypted data issued by geographically dispersed clients. It also provides same elastic availability and scalability of the original cloud DBaaS. So in general, the proposed system provides data confidentiality and security along with cloud database services (availability, elasticity and scalability).

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] Luca Ferretti, Michele Colajanni, and MircoMarchetti , “Distributed, concurrent and Independent Access to Encrypted Cloud Databases”, IEEE Transactions On Parallel and Distributed Systems, Vol. 25, No. 2, February 2014.
[4] 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.
[5] H. Hacigu¨mu¨ s¸, B. Iyer, C. Li, and S. Mehrotra, “Executing SQL over Encrypted Data in the Database-Service-Provider Model,” Proc. ACM SIGMOD Int’l Conf. Management Data, June2002.
[6] E. Mykletun and G. Tsudik, “Aggregation Queries in the Database-as-a-Service Model,” Proc. 20th Ann. IFIP WG 11.3 Working Conf. Data and Applications Security, July/Aug. 2006.
[7] “Transaction Processing Performance Council,” TPC-C, http://www.tpc.org, Apr. 2013.

Author Profile

Anes P.A received the B.Tech (CS) degree from Ilahia College of Engineering and Technology, Muvattupuzha, India in 2006. During 2006-2013, worked as Software Programmer in Web Technologies. In 2013-2015 pursuing M.Tech Computer Science and Engineering Specialization in Cyber Security. His research interests include Ethical hacking, Network security and GUI based software development.

Neethu Francis received the B.Tech (IT) from Viswajyothi College of Engineering in 2012 and M.Tech (CS) from Christ University Faculty Of Engineering, Kengeri, Banglore in 2014. From 2014 November, worked as Assistant Professor in KMP College of Engineering and Technology, Cherukunnam, India. Her research interests include Wireless Sensor Network, Mobile Computing and Computational Intelligence.