International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 12, December 2015
Available at http://internationaljournalofresearch.org

A Novel Approach to Access Encrypted Cloud Databases by Using Distributed and Concurrent Techniques

Polu Gangeswara[1]& Mr. Venkatesh Naik[2]
[1]M.Tech Student, CSE, Chiranjeevi Reddy Institute of Engineering and Technology, Anantapur, AP, INDIA
[2]Assistant Professor & HOD, CSE, Chiranjeevi Reddy Institute of Engineering and Technology, Anantapur, AP, INDIA

Abstract:
The cloud database as a service is novel paradigms that can be support several Internet-based applications, its adoption requires the solution of the information confidentiality problems. We proposed a novel architecture for adaptive encryption of public cloud databases that offers an interesting alternative to the tradeoff between the required data confidentiality level and the flexibility of the cloud database structures at time. We demonstrate the feasibility and performance of the proposed solution through a software prototype. A novel architecture for adaptive encryption of public cloud databases that offers an interesting alternative to the tradeoff between the required data confidentiality level and the flexibility of the cloud database structures at design time. This paper proposes a novel architecture for adaptive encryption of public cloud databases that offers a proxy-free alternative to the system. The project demonstrates the feasibility and performance of the proposed solution through a software prototype. The proposed architecture manages five types of information: plain data represent the tenant information; encrypted data are the encrypted version of the plain data, and are stored in the cloud database; plain metadata represent the additional information that is necessary to execute SQL operations on encrypted data; encrypted metadata are the encrypted version of the plain metadata, and are stored in the cloud database; master key is the encryption key of the encrypted metadata, and is known by legitimate clients.

Keywords: Cloud; Security; Confidentiality; Secured Baas; Database

I.INTRODUCTION

Cloud based mostly services have become common as they specialise in high accessibility and quantifiability at low value. whereas providing high accessibility and quantifiability, inserting essential knowledge to cloud poses several security problems. For avoiding these security problems the info area unit keep within the cloud information in associateencrypted format. The encrypted cloud information permits the execution of SQL operations by choosing the cryptography schemes that support SQL operators. Encrypted cloud information permits differing types of accesses like distributed, concurrent, and freelance. one in all the design that supports these 3 types of access is Secure DBaaS.

The Secure DBaaS design supports multiple and freelance shoppers to execute synchronal SQLoperations on encrypted knowledge. knowledge consistency ought to be maintained by investmentconcurrency management mechanisms utilized in database management system engines. This surveyexplains the assorted concurrency management protocols that may be utilized in the encrypted cloud information. The applications

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 12, December 2015
Available at http://internationaljournalofresearch.org

want 1SR if knowledge is replicated. Hence, to ensure the deserves of cloud, it's essential to produce high quantifiability, accessibility, low value and knowledge with sturdy consistency, that is ready to dynamically adapt to system conditions. Self optimizing one copy serializability (SO-1SR) is that the concurrency management protocol that dynamically optimizes all stages of dealing execution on replicated knowledge within the cloud information. Current DBMSs supported by cloud suppliers permits relaxed consistency guarantees that successively increase the look complexness ofrequests. The second concurrency controllingprotocol is that the pic isolation (SI) that provides redoubled concurrency in cloud atmosphere in comparison to 1SR. Transactions area unit browse from the pic, reads area unit never blocked attributable to write locks that successively will increase concurrency. SI doesn't permit several of the inconsistencies, SI permits dealing inversions. To avoid dealing inversions sturdy consistency guarantee is needed, i.e. sturdy SI (SSI). The third concurrency management protocol is that the session consistency (SC). Session consistency could be a completely different type of ultimate consistency. The system provides browse your writes consistency within every session. Session consistency is at a coffee valuewhereas considering interval and dealing value. The value based mostly concurrency management within the cloud is that the C three i.e. cost-based adaptive concurrency management in cloud. C3 dynamically switch between sturdy consistency level and weak consistency level of transactions during a cloudinformation in line with the value at runtime. it's designed on the highest of 1SR and SSI.

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.

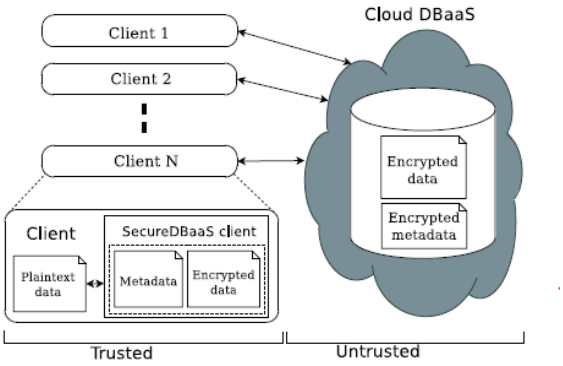

Fig. 1. SecureDBaaS architecture.

III. IMPLEMENTATION

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

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 12, December 2015
Available at http://internationaljournalofresearch.org

and column we have to access it with SQL query only.

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.

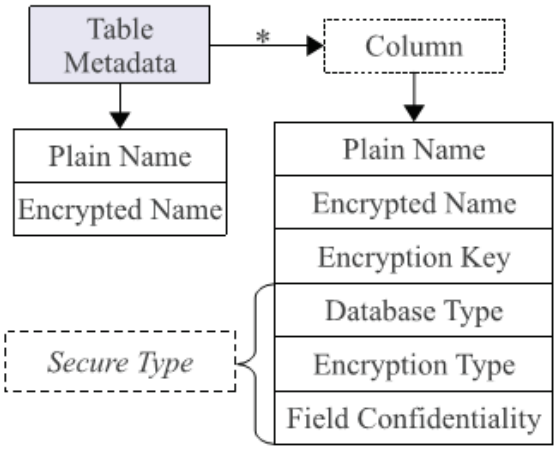

Fig. 2.The structure of a table metadata is represented

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.

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.

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.

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.

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 ofefficient 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

IV. CONCLUSION

We address the data privacy concerns by proposing a novel cloud database model that uses adaptive encryption techniques with no intermediate servers. This scheme provides tenants with the best level of privacy for any database workload that is to change in a medium-term period. We investigate the feasibility and performance of the proposed architecture through a large set of experiments based on a software prototype subject. Our results analysis proved that the cloud networks semantic that are typical of cloud database environments hide most overheads related to static and adaptive encryption. We address the data confidentiality concerns by proposing a novel cloud database architecture that uses adaptive encryption techniques with no intermediate servers. This scheme provides tenants with the best level of confidentiality for any database workload that is likely to change in a medium-term period. We investigate the feasibility and performance of the proposed architecture through a large set of experiments based on a software prototype subject to the TPC-C standard benchmark. Our results demonstrate that the network latencies that are typical of cloud database environments hide most overheads related to static and adaptive encryption. Moreover, we propose a model and a methodology that allow a tenant to estimate the costs of plain and encrypted cloud database services even in the case of workload and cloud price variations in a medium-term horizon. By applying the model to actual cloud provider prices, we can deter- mine the encryption and adaptive encryption costs for data confidentiality. Future research could evaluate the proposed or alternative architectures for multi-user key distribution schemes and under different threat model hypotheses.

V. REFERENCES

[1] Luca Ferretti, Michele Colajanni, and Mirco Marchetti, "Distributed, Concurrent, and Independent Access to Encrypted Cloud Databases", IEEE Transactions on Parallel and Distributed Systems, Vol. 25, No. 2, February 2014.

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

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

[4] 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.

[5] J. Li, M. Krohn, D. Mazieres, and D. Shasha, "Secure Untrusted Data Repository (SUNDR)," Proc. Sixth USENIX Conf. Operating Systems Design and Implementation, Oct. 2004.

[6] 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.

[7] H. Hacigumus¸, B. Iyer, and S. Mehrotra, "Providing Database as a Service," Proc. 18th IEEE Int'l Conf. Data Eng., Feb. 2002.

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

[9] 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.

[10] H. Hacigumus¸, 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, June 2002.

[11] J. Li and E. Omiecinski, "Efficiency and Security Trade-Off in Supporting Range Queries on Encrypted Databases," Proc. 19th Ann. IFIP WG 11.3 Working Conf. Data and Applications Security, Aug. 2005.

[12] 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