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 Secure Architecture for Encipher of Public Cloud Storage

Madhu Babu Balineni[1] & Haik Mohammad Rafi[2]
[1]M-Tech Dept. of CSE Sri Mittapalli Collage of Engineering
Mail Id: -madhu.may19@gmail.com
[2]Associate Professor Dept. of CSE Sri Mittapalli Collage of Engineering
Mail Id: -mdrafi.527@gmail.com

Abstract
Cloud Computing is an emerging accommodation oriented domain which performs Primitive accommodations as IaaS,PaaS and SaaS along with A Cloud database management system (CDBMS) is a distributed database that distributes computing as an accommodation in lieu of a product is called as DBaaS (Database as a Accommodation). The cloud database as an accommodation is a novel paradigm that can fortify several Internet-predicated applications, but its adoption requires the solution of information confidentiality quandaries. In this paper we demonstrate how secure a proposed novel Flexible architecture for encryption of public cloud database as independent encryption architecture which performs double layer encryption in order to for fend Cloud database as well utilizer privacy with data integrity. Adaptive encryption sanctions any sql operation over encrypted data. We can determine the encryption and adaptive encryption cost of data confidentiality from the research perspective. Further we descried to implement optimized technique to perform double layer encryption technique to ameliorate the system performance.

Keywords: Cloud Database; Adaptive Encryption; SQL operations; Metadata; Distributed SQL operation; Multi key Distribution

1. Introduction

In today's world cloud computing has acquired a profoundly and astronomically immense role. It has led to the emergence of an incipient business model kenned as "Database-as-a-service" (DAAS) model [1]. In this model the "Database Accommodation Providers" (DSP) provide database accommodations to customers over the cyber world. DSP exposes its database servers to the client over internet. The client access these database servers utilizing standard SQL interface. Client only has to configure his applications to utilize the database servers hosted by DSP, in lieu of database servers managed locally. The fact that DAAS model is gaining an abundance of favor in industry is evident from emergence of cloud predicated SQL platforms from sundry vendors such as Microsoft SQL Azure, Amazon RDS etc. The elevate in popularity of DAAS is due to numerous benefits it provides to client.

We are fascinated with the database as an accommodation paradigm (DBaaS) that poses several research challenges in terms of security and cost evaluation from a tenant's perspective. Most results concerning encryption for cloud-predicated accommodations are inapplicable to the database paradigm. Other encryption schemes that sanction the execution of SQL operations over encrypted data either have performance limits or require the cull of which encryption scheme must be adopted for each database column and SQL operation [2]. These latter proposals are fine when the set of queries can be statically determined at design time, while we are fascinated with other mundane scenarios where the workload may change after the data- base design. In this paper, we propose a novel architecture for adaptive encryption of public cloud databases that offers a proxy-free alternative to the system described. The proposed architecture guarantees in an adaptive way the best level of data

confidentiality for any database workload, even when the set of SQL queries dynamically changes.

In this paper, we investigate each of these issues and we reach pristine conclusions in terms of prototype implementation, performance evaluation, and cost evaluation. We implement the first proxy-free architecture for adaptive encryption of cloud databases. It does not constrain the availability, elasticity and scalability of a plain cloud database, because concurrent clients can issue parallel operations without passing through some centralized component as in alternative architectures [10]. Moreover, we propose the first analytical cost estimation model for evaluating cloud database costs in plain and encrypted instances from a tenant‟s perspective in a medium-term period [3]. It takes additionally into account the variability of cloud prices and the possibility that the database workload may change during the evaluation period. This model is instanced with deference to several cloud provider offers and cognate authentic prices. As expected, adaptive encryption influences the costs cognate to storage size and network utilization of a database accommodation. However, it is consequential that a tenant can anticipate the final costs in its period of interest, and can cull the best compromise between data confidentiality and expenses.

2. Related Work

An efficacious the privacy of information stored in cloud databases represents a paramount objective to the adoption of the cloud. Our proposal is characterized by two main contributions to the state of the art: architecture and cost model. Albeit data encryption seems the most intuitive solution for privacy, its application to cloud database accommodations is not frivolous, because the cloud database must be able to execute SQL operations directly over encrypted data without accessing any decryption key [7] [8]. The encrypt whole database through some standard encryption algorithms that do not sanction to execute any SQL operation directly on the cloud. As a consequence, the tenant has two alternatives: download the entire database, decrypt it, execute the query and, if the operation modifies the database, encrypt and upload the incipient data; decrypt ephemerally the cloud database, execute the query, and reencrypt it. The former solution is affected by astronomically immense communication and computation overheads, and consequent costs that would make cloud database accommodations quite inconvenient; the latter solution does not assure data confidentiality because the cloud provider obtains decryption keys [6].

The cloud computing paradigm is prosperously converging as the fifth utility , but this positive trend is partially circumscribed by concerns about information confidentiality and obscure costs over a medium-long term .We are intrigued with the Database as a Accommodation paradigm (DBaaS) that poses several research challenges in terms of security and cost evaluation from a tenant's perspective. Most results concerning encryption for cloud-predicated accommodations are in applicable to the database paradigm. Other encryption schemes, which sanction the execution of SQL operations over encrypted data, either suffer from performance limits or they require the cull of which encryption scheme must be adopted for each database column and SQL operations.

Proposed System:

Where in the Proposed System Data owner can upload data in to Cloud database, as a owner afore to upload the data he need to register with policy as Minuscule or Medium or Long And withal provide categorical attributes as access policy in order to restrict access from un sanctioned users , anon after getting registration with cloud server , owner can upload double layer encrypted files in to cloud database , as a cloud server can't read that file , data will be available as encrypted formatted as cipher text. So by providing double layer encryption it doesn't sanction to read and indite to the cloud accommodation providers. When a

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

utilizer want to access cloud database data , he need to be match with access policy with data owner in terms of tenet policy in order to access the data. In this regards we used Blow fish encryption algorithms for layer-1 Encryption and SHA-1 (Secured Hashing Algorithm) is utilized for 2-layer encryption.

The proposed system fortifies adaptive encryption method for public cloud database accommodation, where distributed and concurrent clients can issue direct SQL operations. By eschewing an architecture predicated on one [10] or multiple intermediate servers between the clients and the cloud database, the proposed solution guarantees the same level of scalability and availability of the cloud accommodation. Figure 1 shows a scheme of the proposed architecture where each client executes an encryption engine that manages encryption operations. This software module is accessed by external utilizer applications through the encrypted database interface. The proposed architecture manages five types of information.

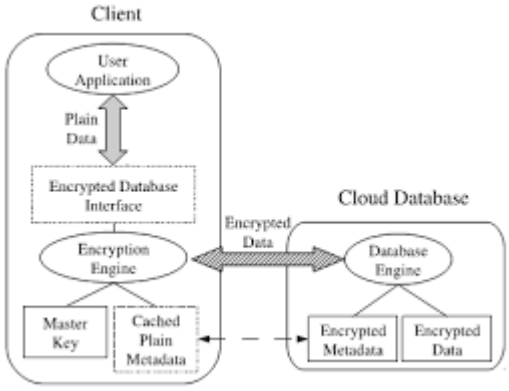

Fig 1: System Architecture.

All data and metadata stored in the cloud database are encrypted. Any application running on a legitimate client can transparently issue SQL operations (e.g., CULL, INSERT, UPDATE and EFFACE) to the encrypted cloud database through the encrypted database interface. Data transferred between the utilizer application and the encryption engines are in plain format, whereas information is always encrypted afore sending it to the cloud database. When an application issues an incipient SQL operation, the encrypted database interface contacts the encryption engine that retrieves the encrypted metadata and decrypts it through the master key [7]. In order to amend performance, the plain metadata are cached locally by the client as volatile information. After obtaining the metadata, the encryption engine is able to execute the SQL operation on encrypted data, and then to decrypt the results. The results are returned to the utilizer application through the encrypted database interface.

3. Implementation

3.1 Flexible encryption:

The proposed system fortifies flexible encryption methods for public cloud database accommodation, where distributed and concurrent clients can issue direct SQL operations. By evading an architecture predicated on one [or] multiple intermediate servers between the clients and the cloud database, the proposed solution guarantees the same level of scalability and availability of the cloud accommodation. Figure 1 show a scheme of the proposed architecture where each client executes an encryption engine that manages encryption operations. This software module is accessed by external utilizer applications through the encrypted database interface. The proposed architecture manages five types of information.

- Plain data is the tenant information;
- Encrypted data is stored in the cloud database;
- Plain metadata represent the supplemental information that is obligatory to execute SQL operations on encrypted data.
- Encrypted metadata is the encrypted version of the metadata that are stored in the cloud database;
- Master key is the encryption key of the encrypted metadata that is distributed to legitimate clients.

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

3.2 Metadata Schema:

Metadata include all information that sanctions a legitimate client kenning the master key to execute SQL operations over an encrypted database. They are organized and stored at a table-level granularity to reduce communication overhead for retrieval, and to ameliorate management of concurrent SQL operations. We define all metadata information associated to a table as table metadata. Let us describe the schema of a table metadata .Table metadata includes the correspondence between the plain table name and the encrypted table name because each encrypted table designation is desultorily engendered [5]. Moreover, for each column of the pristine plain table it additionally includes a column metadata parameter containing the denomination and the data type of the corresponding plain column (e.g., integer, string, timestamp). Each column metadata is associated to one or more onion metadata, as many as the number of onions cognate to the column.

3.3 Encrypted database directorate:

The database administrator engenders a master key, and utilizes it to initialize the architecture metadata. The master key is then distributed to legitimate clients. Each table engenderment requires the insertion of an incipient row in the metadata table. For each table engenderment, the administrator integrates a column by designating the column designation, data type and confidentiality parameters. These last are the most consequential for this paper because they include the set of onions to be associated with the column, the commencement layer (denoting the genuine layer at engenderment time) and the field confidentiality of each onion. If the administrator does not designate the confidentiality parameters of a column, then they are automatically culled by the client with veneration to a tenant's policy [8]. Typically, the default policy postulates that the commencement layer of each onion is set to its most vigorous encryption algorithm.

3.4 Cost Estimation of cloud database services:

A tenant that is fascinated with estimating the cost of porting its database to a cloud platform. This porting is a strategic decision that must evaluate confidentiality issues and the cognate costs over a medium-long term. For these reasons, we propose a model that includes the overhead of encryption schemes and variability of database workload and cloud prices. The proposed model is general enough to be applied to the most popular cloud database accommodations, such as Amazon Relational Database Accommodation.

3.5 Cost model:

The cost of a cloud database accommodation can be estimated as a function of three main parameters:

Cost = f(T ime, Pricing,Utilization) where:

• Time: identifies the time interval T for which the tenant requires the accommodation.

• Pricing: refers to the prices of the cloud provider for subscription and resource utilization; they typically incline to diminish during T.

• Utilization: denotes the total amount of resources utilized by the tenant; it typically increases during T .In order to detail the pricing attribute, it is consequential to designate that cloud providers adopt two subscriptions.

3.6 Policies:

The on-demand policy sanctions a tenant to paper-use and to withdraw its subscription anytime; the reservation policy requires the tenant to commit in advance for a reservation period. Hence, we distinguish between billing costs depending on resource utilization and reservation costs denoting supplemental fees for commitment in exchange for lower pay-per-use prices. Billing costs are billed periodically to the tenant every billing period.

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

3.6 Cloud pricing models:

Popular cloud database providers adopt two different billing functions, that we call linear L and tiered T . Let us consider a generic resource x, we define as xb its utilization at the b-th billing period and px b its price. If the billing function is tiered, the cloud provider uses different prices for different ranges of resource utilization. Let us define Z as the number of tiers, and [ˆx1, . . . , ˆxZ−1] as the set of thresholds that define all the tiers. The uptime and the storage billing functions of Amazon RDS are linear, while the network utilization is a tiered billing function. On the other hand, the uptime billing functions of Azure SQL is linear, while the storage and network billing functions are tiered.

3.7 Usage Estimation:

The uptime is facilely quantifiable, it is more arduous to estimate accurately the utilization of storage and network , since they depend on the database structure, the workload and the utilization of encryption. We now propose a methodology for the estimation of storage and network utilization due to encryption. For pellucidity, we define sp, se, sa as the storage utilization in the plaintext, encrypted, and adaptively encrypted databases for one billing period. Similarly, np, ne, na represent network utilization of the three configurations. We postulate that the tenant kens the database structure and the query workload and we postulate that each column aA stores ra values. By denoting as vp a the average storage size of each plaintext value stored in column a, we estimate the storage of the plaintext database.

4. Experimental Results

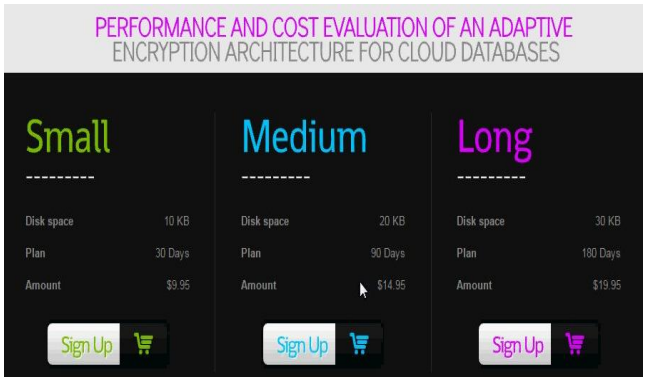

Fig 2: Database Partitions considered as Cost.

Fig 3: For Secure Data Stored with encrypted Format.

5. Conclusion

In this paper we demonstrate how secure a proposed novel Flexible architecture for encryption of public cloud database as independent encryption architecture which performs double layer encryption in order to for fend Cloud database as well utilizer privacy with data integrity. Adaptive encryption sanctions any sql operation over encrypted data. The novel cloud database architecture that utilizes adaptive encryption technique with no intermediate servers. This scheme provides cloud provider with the best level of confidentiality for any database workload. We can determine the encryption and adaptive encryption cost of data confidentiality from the research perspective. Further we descried to implement optimized technique to perform double layer encryption technique to ameliorate the system performance.

6. References

[1] H. Hacigumus¸, B. Iyer, C. Li, and S. Mehrotra, "Executing SQL over Encrypted Data in the DatabaseService-Provider Model," Proc. ACM SIGMOD Int'l Conf. Management Data, June 2002.

[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 DatabaseService-Provider Model," Proc. ACM SIGMOD Int'l Conf. Management Data, June 2002.

Authors Profile

HAIK MOHAMMAD RAFI

Received B-Tech (comp) from Jawaharlal Nehru Technological University,
M-Tech (comp) from AcharyaNagarjunaUniversity. Pursuing PhD from AcharyaNagarjunaUniversityPresently working asAssociate Professor in Sri MittapalliCollege of engineering, affiliated to J.N.T.U,Kakinada. My area of interest is SoftwareReliability, Software Architecture Recovery,Network Security, and Software Engineering. Email:-mdrafi.527@gmail.com