Scientific Journal Impact Factor (SJIF) : 3.518

International Journal of Modern Trends in Engineering and Research

ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

A Review on Evaluation of an Adaptive Encryption Architecture for Cloud Databases: Performance and Cost Perspective

Ms. Sarika S Malani[1] and Prof. Ganesh B.Regulwar[2]

[1] *ME second Year CSE, BNCOE, Pusad*

[2] *Assistant Professor, BNCOE, Pusad*

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. We propose an original cost model that is oriented to the evaluation of cloud database services in plain text and encrypted instances and that takes into account the variability of cloud prices and tenant workloads during a medium-term period.

Keywords- adaptively, cloud database, cost model, confidentiality, encryption

I. INTRODUCTION

The cloud computing paradigm is successfully converging as the fifth utility [1], but this positive trend is partially limited by concerns about information confidentiality [2] and unclear costs over a medium-long term [3], [4]. We are interested in the database as a service paradigm (DBaaS) [5] that poses several research challenges in terms of security and cost evaluation from a tenant's point of view. Most results concerning encryption for cloud-based services [6], [7] are inapplicable to the database paradigm. Other encryption schemes that allow the execution of SQL operations over encrypted data either have performance limits [8] or require the choice of which encryption scheme must be adopted for each database column and SQL operation [9]. These latter proposals are fine when the set of queries can be statically determined at design time, while we are interested in other common scenarios where the workload may change after the database 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 in [10]. 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.

The proposed system supports adaptive encryption methods for public cloud database service, where distributed and concurrent clients can issue direct SQL operations. By avoiding an architecture based 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 service.

II. LITERATURE SURVEY

2.1. Compostable cost estimation and monitoring for computational applications in cloud computing environments

With the from cloud computing providers, scientists have the opportunity to utilize pay-as-you-go resources together with their own and shared resources. However, scientists need to decide which parts of their applications should be executed in cloud computing systems in order to balance the trade-o_ between cost, time and resource requirements. In this paper, we present a service for estimating, monitoring and analyzing costs associated with scientific applications in the cloud. Cost models associated with deferent application execution models are proposed and these cost models can be composed to determine costs of deferent scenarios. We present techniques to estimate costs for service dependency and to monitor costs associated with typical scientific applications. Experiments with real-world applications are performed to illustrate the usefulness of our techniques. Our service could eventually be integrated into cloud resource management and execution services to support on-the-fly resource scheduling. Recently, cloud computing has been considered as an emerging model which aims at allowing customers to utilize computational resources and software hosted by service providers [1, 2, 3], thus shifting the complex and tedious resource and software management tasks typically done by the customers to the service providers. Cloud computing promises to eliminate obstacles due to the management of IT resources and to reduce the cost on infrastructure investments. As a result, besides business customers, cloud computing is also attractive to many scientists from small research groups in the computational science and engineering (CSE) field. However, still there are many unclear questions about how cloud computing can help CSE scientists [3, 1]. Among them, the need to determine the actual cost when using cloud computing is evident. In general, the cost determination is necessary for investigating the return of investment, and, in particular, it is needed to decide when and under which forms cloud computing offers can be used. Let us consider this particular point in the interest of small CSE research groups which have limited resources or limited access to their shared computational resources and storages. These groups cannot entirely rely on that resources and storages as well as on cloud computing offers due to several reasons. Scientists of these groups need a quick assertion on the cost of executing their applications in the cloud. They want to evaluate if it makes sense to run a particular application or parts of the application using cloud computing, if cloud resources should be used in a regular or occasional basis, and if all resources of need are fully or partially based on clouds. Using information provided by the vendor, it is very midcult to calculate the cost of an application because scientists need to map the computation and data transfer requirements associated with their CSE applications to primitive prices of CPUs, storages and network transfer. Scientists expect to have cost models associated with application models, e.g., Open, MPI, and workflows. Furthermore, a scientific experiment might include deferent parts, each may have a deferent application model and might or might not be executed in the cloud. Thus, it is interesting to have compostable cost models in order to decide which parts of the experiment should be executed by using cloud resources. In this paper, we present a service for estimating and monitoring costs associated with CSE applications that is, in particular, suitable for scientists from small CSE groups. These scientists have some particular constraints that require them to use deferent resources in deferent ways based on fully on-premise, partially cloud, and fully cloud resources. We present compostable cost models. Based on these models, we develop a service which supports both cost estimation and real-time monitoring of costs for CSE applications in the cloud. [3]

2.2. Providing Database as a Service

we explore a new paradigm for data management in which a third party service provider hosts ”database as a service” providing its customers seamless mechanisms to create, store, and access their databases at the host site. Such a model alleviates the need for organizations to purchase expensive hardware and software, deal with software upgrades, and hire professionals for administrative and maintenance tasks which are taken over by the service provider. We have developed and deployed a database service on the Internet, called NetDB2, which is in constant use. In a sense, data management model supported by NetDB2 provides an effective mechanism for organizations to purchase data management as a service, thereby freeing them to concentrate on their core businesses. Among the primary challenges introduced by"database as a service" are additional overhead of remote access to data, an infrastructure to guarantee data privacy, and user interface design for such a service. These issues are investigated in the study. We identify data privacy as a particularly vital problem and propose alternative solutions based on data encryption. This paper is meant as a challenges paper for the database community to explore a rich set of research issues that arise in developing such a service. Advances in the networking technologies have triggered one of the key industry responses, the"software as a service" initiative, also referred to as the application service provider (ASP) model. In this paper, we explore the"database as service" paradigm and the challenges introduced by that. Today, efficient data processing is a fundamental and vital issue for almost every scientific, academic, or business organization. Therefore the organizations end up installing and managing database management systems to satisfy different data processing needs. Although it is possible to purchase the necessary hardware, deploy database products, establish network connectivity, and hire the professional people who run the system, as a traditional solution, this solution has been getting increasingly expensive and impractical as the database systems and problems become larger and more complicated. As it is described above, the traditional solution entails different costs. It might be arguable that hardware, software, and network costs are decreasing constantly. People costs, however, generally, do not decrease. In the future, it is likely that computing solution costs will be dominated by people costs. There is need for database backup, database restore, and database reorganization to reclaim space or to restore preferable arrangement of data. Migration from one database version to the next, without impacting solution availability, is an art still in its infancy [5]. Parts of a database solution, if not the entire solution usually become unavailable during version change. An organization that provides database service has an opportunity to do these tasks and offer a value proposition provided it is efficient. The new paradigm challenges the traditional model of data management followed by current organizations. Database service provider provides seamless mechanisms for organizations to create, store, and access their databases. Moreover, the entire responsibility of database management, i.e., database backup, administration, restoration, and database reorganization to reclaim space or to restore preferable arrangement of data, migration from one database version to the next without impacting availability will befall such an organization. Users wishing to access data will now access it using the hardware and software at the service provider instead of their own organization's computing infrastructure. The application would not be impacted by outages due to software, hardware and networking changes or failures at the database service provider's site. This would alleviate the problem of purchasing, installing, maintaining and updating the software and administrating the system. Instead of doing these, the organization will only use the ready system maintained by the service provider for its database needs. The technological aspects of developing database as a service lead to new research challenges. First and foremost is the issue of data privacy. In the database service provider model, user data needs to reside on the premises of the database service provider. Most corporations view their data as a very valuable asset. The service provider would need to provide sufficient security measures to guard the data privacy. We propose data encryption as the solution to this problem. Second key challenge is that of performance. Since the interaction between the users and the database service provider takes place in a different medium, the network, than it does in traditional databases, there are potential overheads introduced by this architecture. Therefore the sources of performance degradation and its significance should be determined. Another challenge facing the database service provider model is that of an appropriate user interface. Clearly, the interface must be easy to use; yet it needs to be powerful enough to allow ease in building applications. We have developed and deployed a database service on the Internet, called NetDB2, an experimental network based application service provider (ASP) system. It has been operational over a year and used by number of universities to help teaching database courses at different locations. NetDB2 provides database services including tools for application development, creating and loading tables, and performing queries and transactions to the users over the Internet. [5]

2.3. Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services

Cloud computing, as an emerging computing paradigm, enables users to remotely store their data into a cloud so as to enjoy scalable services on-demand. Especially for small and medium-sized enterprises with limited budgets, they can achieve cost savings and productivity enhancements by using cloud-based services to manage projects, to make collaborations, and the like. However, allowing cloud service providers (CSPs), which are not in the same trusted domains as enterprise users, to take care of confidential data, may raise potential security and privacy issues. To keep the sensitive user data confidential against entrusted CSPs, a natural way is to apply cryptographic approaches, by disclosing decryption keys only to authorized users. However, when enterprise users outsource confidential data for sharing on cloud servers, the adopted encryption system should not only support fine-grained access control, but also provide high performance, full delegation, and scalability, so as to best serve the needs of accessing data anytime and anywhere, delegating within enterprises, and achieving a dynamic set of users. In this paper, we propose a scheme to help enterprises to efficiently share confidential data on cloud servers. We achieve this goal by first combining the hierarchical identity-based encryption (HIBE) system and the cipher text-policy attribute-based encryption (CP-ABE) system, and then making a performance-expressivity tradeoff, finally applying proxy re-encryption and lazy re-encryption to our scheme. With the emergence of sharing confidential corporate data on cloud servers, it is imperative to adopt an efficient encryption system with a fine-grained access control to encrypt outsourced data. Ciphertext-policy attribute-based encryption (CP-ABE), as one of the most promising encryption systems in this field, allows the encryption of data by specifying an access control policy over attributes, so that only users with a set of attributes satisfying this policy can decrypt the corresponding data. However, a CP-ABE system may not work well when enterprise users outsource their data for sharing on cloud servers, due to the following reasons: First, one of the biggest merits of cloud computing is that users can access data stored in the cloud anytime and anywhere using any device, such as thin clients with limited bandwidth, CPU, and memory capabilities. Therefore, the encryption system should provide high performance. Second, in the case of a large-scale industry, a delegation mechanism in the generation of keys inside an enterprise is needed.

Although some CP-ABE schemes support delegation between users, which enables a user to generate attribute secret keys containing a subset of his own attribute secret keys for other users, we hope to achieve a full delegation, that is, a delegation mechanism between attribute authorities (AAs), which independently make decisions on the structure and semantics of their attributes. Third, in case of a large-scale industry with a high turnover rate, a scalable revocation mechanism is a must. The existing CP-ABE schemes usually demand users to heavily depend on AAs and maintain a large amount of secret keys storage, which lacks flexibility and scalability.

Our main design goal is to help the enterprise users to efficiently share confidential data on cloud servers. Specifically, we want to make our scheme more applicable in cloud computing by simultaneously achieving fine-grained access control, high performance, practicability, and scalability.

In this paper, proposed a hierarchical attribute-based encryption (HABE) model by combining a HIBE system and a CP-ABE system, to provide fine-grained access control and full delegation. Based on the HABE model, we construct a HABE scheme by making a performance-expressivity tradeoff, to achieve high performance. Finally, we propose a scalable revocation scheme by delegating to the CSP most of the computing tasks in revocation, to achieve a dynamic set of users efficiently. [6]

2.4. Fully Homomorphism Encryption Using Ideal Lattices

We propose a fully homomorphism encryption scheme – i.e., a scheme that allows one to evaluate circuits over encrypted data without being able to decrypt. Our solution comes in three steps. First, we provide a general result – that, to construct an encryption scheme that permits evaluation of arbitrary circuits, it suffices to construct an encryption scheme that can evaluate (slightly augmented versions of) its own decryption circuit; we call a scheme that can evaluate its (augmented) decryption circuit boots trappable. Next, we describe a public key encryption scheme using ideal lattices that is almost boot strippable. Lattice-based cryptosystems typically have decryption algorithms with low circuit complexity, often dominated by an inner product computation that is in NC1. Also, ideal lattices provide both additive and multiplicative homeomorphisms (modulo a public-key ideal in a polynomial ring that is represented as a lattice), as needed to evaluate general circuits. Unfortunately, our initial scheme is not quite bootstrappable – i.e., the depth that the scheme can correctly evalu- ate can be logarithmic in the lattice dimension, just like the depth of the decryption circuit, but the latter is greater than the former. In the final step, we show how to modify the scheme to reduce the depth of the decryption circuit, and thereby obtain a boot strippable encryption scheme, with- out reducing the depth that the scheme can evaluate. Abstractly, we accomplish this by enabling the encrypter to start the decryption process, leaving less work for the decrypter, much like the server leaves less work for the de- creeper in a server-aided cryptosystem. We propose a solution to the old open problem of constructing a fully homomorphism encryption scheme. [11]

III. PROPOSED SYSTEM

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. The adaptive encryption scheme, which was initially proposed for applications not referring to the cloud, encrypts each plain column to multiple encrypted columns, and each value is encapsulated in different layers of encryption, so that the outer layers guarantee higher confidentiality but support fewer computation capabilities with respect to the inner layers. The outer layers are dynamically adapted at runtime when new SQL operations are added to the workload. Although this adaptive encryption architecture is attractive because it does not require to define at design time which database operations are allowed on each column, it poses novel issues in terms of applicability to a cloud context, and doubts about storage and network costs. We investigate each of these issues and we reach three original conclusions in terms of prototype implementation, performance evaluation, and cost evaluation. We initially design the first proxy-free architecture for adaptive encryption of cloud databases that does not limit the availability, elasticity and scalability of a plain cloud database because multiple clients can issue concurrent operations without passing through some centralized component as in alternative architectures. Then, we evaluate the performance of encrypted database services by assuming the standard TPC-C benchmark as the workload and by considering different network latencies. Thanks to this test bed, we show that most performance overheads of adaptively encrypted cloud databases are masked by network latencies that are typical of a geographically distributed cloud scenario.

we propose the first analytical cost estimation model for evaluating cloud database costs in plaintext and encrypted configurations from a tenant's point of view over a medium-term period. This model also considers the variability of cloud prices and of the database workload during the evaluation period, and allows a tenant to observe how adaptive encryption influences the costs related to storage and network usage of a database service. By applying the model to several cloud provider offers and related prices, the tenant can choose the best compromise between the data confidentiality level and consequent costs in his period of interest.

There are two main tenant concerns that may prevent the adoption of the cloud as the fifth utility: data confidentiality and costs. In this we address both issues in the case of cloud database services. These applications have not yet received adequate attention by the academic literature, but they are of utmost importance if we consider that almost all important services are based on one or multiple databases.

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 will 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 mid-term horizon. By instantiating the model with actual cloud provider prices, we can determine the encryption and adaptive encryption cost of data confidentiality. From the research point of view, it would be also interesting to evaluate the proposed or alternative architectures under different threat model hypotheses.

The proposed architecture guarantees data confidentiality in a security model in which: the network is untrusted; tenant users are trusted, that is, they do not reveal information about plain data, plain metadata, and the master key; the cloud provider administrators are defined semi-honest, that is, they do not modify tenant's data and results of SQL operations, but they could be interested in accessing tenant's information stored in the cloud database.

IV. ARCHITECTURE

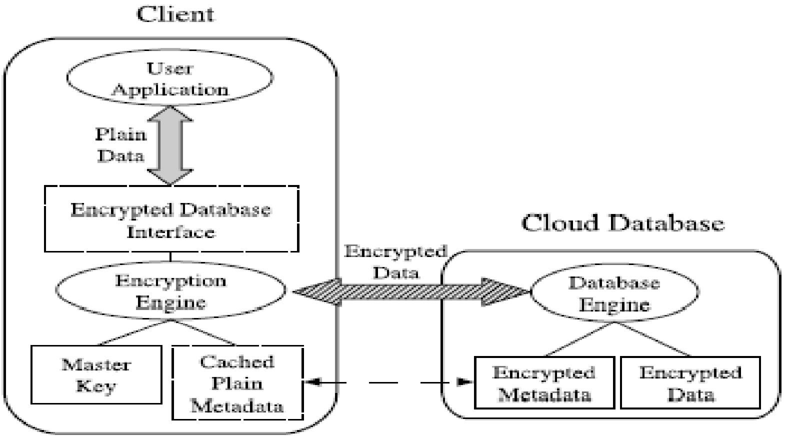

Figure 1. Encrypted cloud database architecture

Modules description:
4.1 System Model
4.2 Adaptive Encryption Scheme
4.3 Cost estimation of cloud database services
4.4 Performance Evolution

4.1. System Model

The proposed system supports adaptive encryption for public cloud database services, where distributed and concurrent clients can issue direct SQL operations. By avoiding an architecture based on intermediate servers between the clients and the cloud database, the proposed solution guarantees the same level of scalability and availability of the cloud service. A scheme of the proposed architecture where each client executes an encryption engine that manages encryption operations. This software module is accessed by external user applications through the encrypted database interface. The proposed architecture manages five types of information:

1. Plain data represent the tenant information
2. Encrypted data are the encrypted version of the plain data, and are stored in the cloud database;
3. Plain metadata represent the additional information that is necessary to execute SQL operations on encrypted data
4. Encrypted metadata are the encrypted version of the plain metadata, and are stored in the cloud database;
5. Master key is the encryption key of the encrypted metadata, and is known by legitimate clients.

4.2. Adaptive Encryption Scheme

We consider SQL-aware encryption algorithms that guarantee data confidentiality and allow the cloud database engine to execute SQL operations over encrypted data. As each algorithm supports a specific subset of SQL operators, we refer to the following encryption schemes.

1. Random (Rand): it is the most secure encryption because it does not reveal any information about the original plain value (IND-CPA). It does not support any SQL operator, and it is used only for data retrieval.
2. Deterministic (Det): it deterministically encrypts data, so that equality of plaintext data is preserved. It supports the equality operator.
3. Order Preserving Encryption (Ope) [12]: it preserves in the encrypted values the numerical order of the original unencrypted data. It supports the comparison SQL operators (i.e., =; <;<_; >;>_).
4. Homomorphic Sum (Sum) [13]: it is homomorphic with respect to the sum operation, so that the multiplication of encrypted integers is equal to the sum of plaintext integers. It supports the sum operator between integer values.
5. Search: it supports equality check on full strings (i.e., the LIKE operator).
6. Plain: it does not encrypt data, but it is useful to support all SQL operators on non confidential data.

If each column of the database was encrypted through only one algorithm, then the database administrator would have to decide at design time which operations must be supported on each database column. However, this solution is impractical for scenarios in which the database workload changes over time.

As an example, consider a database supporting a web application for which feature or security updates are released. If the updates introduce new SQL operations that were not considered at database design time, data encryption will prevent their execution. Moreover, a similar approach prevents data analytics on an encrypted database, since all the queries that were not considered at database design time cannot be executed. Finally, if the encrypted database is not subject to a well-defined workload (e.g. because it is queried by employees, rather than by applications) this encryption strategy cannot be applied.

These issues can be addressed through adaptive encryption schemes that support at runtime any SQL operation while preserving the maximum level of data confidentiality on the columns that are never involved in any operation.

4.3. Cost estimation of cloud database services

We consider a tenant that is interested in estimating the cost of porting his database to a cloud platform. This porting is a strategic decision that must evaluate confidentiality issues and related costs over a medium-long term. For these reasons, we propose a model that includes the overhead of encryption schemes and the variability of database workload and cloud prices. The proposed model is general enough to be applied to the most popular cloud database services, such as Amazon Relational Database Service, Enterprise DB, Windows Azure SQL Database, and Rack space Cloud Database.

4.4. Performance Evolution

This section aims to verify whether the overheads of adaptive encryption represent an acceptable compromise between performance and data confidentiality for the tenants of cloud database services. To this purpose, we design a suite of performance tests that allow us to evaluate the impact of encryption and adaptive encryption on response times and throughput for different network latencies and for increasing numbers of concurrent clients.

V. CONCLUSION

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.

REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, “Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility,” Future Generation Comput. System, volume. 25, no. 6, pp. 599–616, 2009.

[2] T. Mather, S. Kumaraswamy, and S. Latif, “Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance.” Sebastopol, CA, USA:O’Reilly Media, 2009.

[3] H.-L. Truong and S. Dustdar, “Composable cost estimation and monitoring for computational applications in cloud computing environments,” Procedia Comput. Science, volume. 1, no. 1, pp. 2175–2184, 2010.

[4] E. Deelman, G. Singh, M. Livny, B. Berriman, and J. Good, “The cost of doing science on the cloud: The montage example,” in Proceeding. ACM/IEEE Conference. Supercomputing, 2008, pp. 1–12.

[5] H. Hacig€um€u¸s, B. Iyer, and S. Mehrotra, “Providing database as a service,” in Proceeding. 18th IEEE International. Conference. Data Enggnering., Feb. 2002, pp. 29–38.

[6] G. Wang, Q. Liu, and J. Wu, “Hierarchical attribute-based encryption for fine-grained access control in cloud storage services,” in Proceeding. 17th ACM Conference. Computer and Communications Security, 2010, pp. 735–737.

[7] Google. (2014, Mar.). Google Cloud Platform Storage with serverside encryption [Online]. Available: blogspot.it/2013/08/google-cloud-storage-now provides.html.

[8] H. Hacig€um€u¸s , B. Iyer, C. Li, and S. Mehrotra, “Executing SQL over encrypted data in the database-service-provider model,” in Proceeding. ACMSIGMODInt’l Conference. Manage. Data, Jun. 2002, pp. 216–227.

[9] L. Ferretti,s M. Colajanni, and M. Marchetti, “Distributed, concurrent, and independent access to encrypted cloud databases,” IEEE Trans. Parallel Distribution. System, volume. 25, no. 2, pp. 437–446, Feb. 2014.

[10]R.A.Popa, C.M.S.Redfield, N.Zeldovich, and H.Balakrishnan,“CryptDB: Protecting confidentiality with encrypted query processing,” in Proceeding. 23rd ACM . Operating Systems Principles, Oct. 2011, pp. 85–100.

[11] C.Gentry,“Fully homomorphic encryption using ideal lattices,” in Proceeding. 41st ACM. Theory of computing, May 2009

[12] A. Boldyreva , N. Chenette and A. O'Neill,” Order-Preserving Encryption Revisited: Improved Security Analysis and Alternative Solutions” in Proceeding. Advances in Cryptology – CRYPTO 2011.Springer, Aug. 2011.

[13] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in Proc. Advances in Cryptology – EUROCRYPT99.Springer, May 1999.