International Journal of Science Engineering and Advance Technology

Evaluating Cost and Performance Of Adaptive Encryption Architecture for Cloud Database

[1]MD. Rafi Yuddin, [2] Mr .Sheik Ahmad Shah

[1]M.tech Student, [2]Assistant professor (CSE) in Kakinada Institute Of Engineering And Technolgy, Korangi

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.

I. Introduction

The cloud computing paradigm is successfully converge 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 data- base 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 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 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 dy- namically changes. The adaptive encryp- tion 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 en- cryption, so that the outer layers guarantee higher confidentiality but support fewer com- potation capabilities with respect to the inner layers.
The use of fully homomorphism encryption [11] would guarantee the execution of any operation over encrypted data, but existing implementa- tions are affected by huge computational costs to the extent that the execution of SQL opera- tions over a cloud database would become impractical. Other encryption algorithms cha- racterized by acceptable computational com- plexity support a subset of SQL opera- tors [12], [13], [14]. For example, an encryption algorithm may support the order comparison command [12], but not a search operator [14]. The drawback related to these feasible encryption algorithms is that in a medium-long term horizon, the database administrator cannot know at design time which database operations will be required over each database column. This issue is in part addressed in [10] by pro- posing an adaptive encryption architecture that is founded on an intermediate and trusted proxy.

II. Related work

Improving the confidentiality of information stored in cloud databases represents an important contribution to the adoption of the cloud as the fifth utility because it addresses most user concerns. Our proposal is characterized by two main contributions to the state of the art: architecture and cost model. Although data encryption seems the most intuitive solution for confidentiality, its application to cloud data- base services is not trivial, because the cloud database must be able to execute SQL operations directly over encrypted data without accessing any decryption key.

Native solutions encrypt the whole database through some standard encryption algorithms that do not allow 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 data- base, encrypt and upload the new data; decrypt temporarily the cloud database, execute the query, and re-encrypt it. The former solution is affected by huge communication and computation over- heads, and consequent costs that would make cloud database services quite inconvenient; the latter solution does not guarantee data confidentiality because the cloud provider obtains decryption keys.

The right alternative is to execute SQL opera- tions directly on the cloud database, without giving decryption keys to the provider. An ini- tial solution presented in [5] is based on data aggregation techniques [8],that associate plain- text metadata to sets of encrypted data. However, plaintext metadata may leak sensitive information and data aggregation introduces unnecessary network overheads.

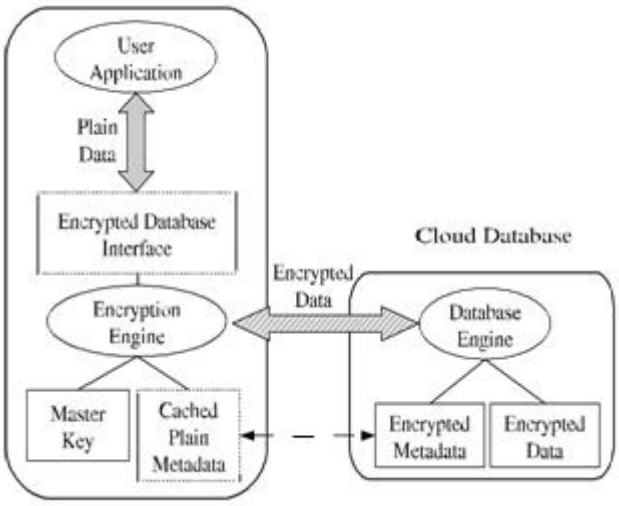

Fig. 1.Encrypted cloud database architecture.

The proposed system supports adaptive encryp- tion for public cloud database services, where distributed and concur- rent clients can issue direct SQL operations. By avoiding an archi- tecture based on intermediate servers [10] be- tween the clients and the cloud database, the proposed solution guarantees the same level of scalability and availability of the cloud service. Fig. 1 shows a scheme of the proposed archi- tecture where each client executes an encryp- tion 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.
All data and metadata stored in the cloud da- tabase are encrypted. Any application running on a legitimate client can transparently issue SQL operations (e.g., SELECT, INSERT, UPDATE and DELETE) to the encrypted cloud database through the encrypted database interface. Data transferred between the user application and the encryption engine are not encrypted, whereas information is always encrypted before sending it to the cloud database. When an application issues a new SQL operation, the encrypted data- base interface contacts the encryption enginethat retrieves the encrypted metadata and de- crypts them with the master key. To improve performance, the plain metadata are cached locally by the client. After obtaining the me- tadata, the encryption engine is able to issue encrypted SQL statements to the cloud database, and then to decrypt the results. The results are returned to the user application through the en- crypted database interface.

2.1 Adaptive Encryption Schemes

We consider SQL-aware encryption algorithms that guarantee data confidentiality and allow the cloud database engine to execute SQL opera- tions over encrypted data. As each algorithm supports a specific subset of SQL operators, we refer to the following encryption schemes.Outer layers guarantee higher level of data confiden- tiality and support fewer operations on encrypted data. Hence, each onion supports a specific set of operations.
Its onions. For example, the plaintext values associated with Onion-Eq are encrypted with Det, then the Det value is encrypted with Rand. The most external layer of an onion is called actual layer, which corresponds to its strongest encryption algorithm. The cloud data- base can only see the actual layer of the onions, and has no access to inner layers nor to plaintext data.

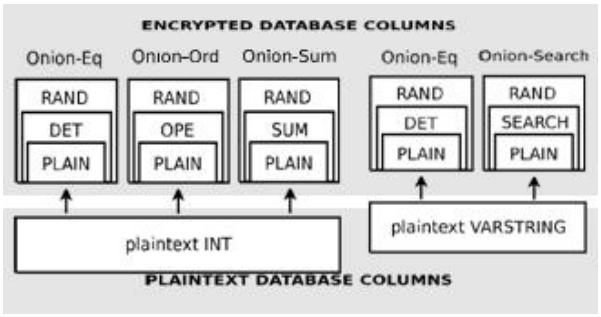

Fig. 2 shows an example of the onions and layers structures by considering two plaintext columns having data types int and varchar. The integer column is encrypted with Onion-Eq, Onion-Ord, and Onion-Sum, and the string col- umn is encrypted with Onion-Eq and Onion-Search. Each onion represents a column in the encrypted database struc- ture. The actual layers of all the onions are set to Rand, thatguarantees the best data confidentiality level but it does not allow computations on encrypted data. When an equality check is requested on the integer column the adaptive layer removal mechanism removes the Rand layer of Onion-Eq, thus leaving Det as its new actual layer.

2.2 Adaptive Layer Removal

The adaptive layer removal is the process that dynamically removes the external layer of an onion in order to adap- tively support SQL operations issued by legitimate clients.

Let us describe the details of the adaptive layer removal mechanism by referring to the following example. We consider a table T with columns id of type int and name of type string, and a tenant client preparing to issue the following statement to the encrypted cloud database: SELECT FROM T WHERE id < 10. The client encryption en- gine analyzes the SQL statement, and iden- tifies that the operation id < 10 has to be executed on the encrypted database. Then, the client reads the metadata and checks whether there is the Onion-Ord attribute associated with the column id because this is the only onion supporting the operator.

If the actual layer of Onion-Ord asso- ciated with id is set to Rand, then the client dynamically invokes a stored procedure on the cloud database that removes at run- time the Rand layer of Onion-Ord of the column id, thus leaving the Ope layer ex- posed. The client can now encrypt the SELECT query that contains the operation id
< 10 and issue the encrypted query to the encrypted database, that executes it on the Ope layer of Onion-Ord. Any new SQL operation involving an order comparison on the column id does not require to invoke again the layer removal procedure be- cause the actual layer of Onion-Ord is Ope
.The cloud database can execute the adaptive layer removal if and only if a legiti- mate client invokes the stored procedure andgives to it the decryption key of the most external encryption layer.

Popular cloud database providers adopt two different bill- ing functions, that we call linear L and tiered T . Let us con- sider a generic re- source x. We define xb as its usage at the b-th billing period and px as its price. If the billing function is linear, it can be computed as:

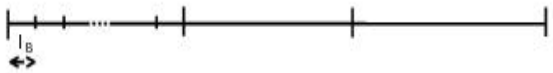

Fig. 4. Example of relationship among estimation (T), reserva- tion (TR) and billing (T_B) periods.

III. Cost estimation of cloud database services

We consider a tenant that is interested in esti- mating 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 va- riability of database work- load and cloud prices. The proposed model is general enough to be applied to the most popular cloud database services, such as Amazon Relational The uptime and the storage billing functions of Amazon RDS [23] are linear, while the net- work usage 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.

The current version of the prototype supports the main SQL operations (SELECT, DELETE, INSERT and UPDATE) and the WHERE clause. We consider three TPC-C compli- ant databases having 10 warehouses: Plaintext (PLAIN) is based on plaintext data. Encrypted (ENC) re- fers to a statically encrypted data- base where each column is encrypted at design time with Database Service Enterprise DB .Adaptively Azure SQL Database and Rackspace Cloud Database.

4.1 Cost Model

The cost of a cloud database service can be estimated as a function of three main parameters Cost ¼ f ð Time; Pricing; Usage Þ; (1)

Where: Time identifies the time interval T for which the ten- ant requires the service. Pricing refers to the prices of the cloud provider for subscription and resource usage they typically tend to diminish dur- ing T . Usage denotes the total amount of re- sources used by the tenant; it typically increases during T . The total cost of the cloud database service C can be estimated through the following equation: encrypted (ADAPT) refers to an encrypted da- tabase in which each column is encrypted with all the onions supported by its data type (Section 3.3). In the ENC and ADAPT configurations each column is set to the highest encryption layer that supports the SQL operations of the TPC-C workload. During each TPC-C test lasting for 300 seconds, we monitor the number of executed TPC-C transactions, and the response times of all the SQL operations from the standard TPC-C work- load. We repeat the test for each database configuration (PLAIN, ENC and ADAPT) for increasing number of cli- ents (from 5 to 20), and for increasing network latencies (from 0 to 120 ms). To guarantee data consistency the three da- tabases use repeatable read (snapshot) isolation level.

IV. Performance evaluation

The experiments aim to evaluate the overhead caused by static and adaptive encryption in terms of system through- put and response time. In Figs. 5 and 6, we report the num- ber of com- mitted TPC-C transactions per minute executed on the three cloud database

configurations for 5 and 20 concurrent clients, respectively. We can appreciate that in both cases, and in all other results not reported for space reasons, the throughput of the ENC database is close to that of the PLAIN database. Moreover, as the network latency increases, even the performance of the ADAPT database tends to that of the other two configurations.

W	Estimated Storage [MB] (Error %)		
	PLAIN	ENC	ADAPT
1	99 (1.0)	187 (5.6)	273 (6.6)
5	453 (1.6)	859 (5.4)	1270 (6.3)
10	894 (1.5)	1698 (5.3)	2516 (6.2)

	Estimated	Measured	Error
ENC	13175	13329	-1.2%
ADAPT	13671	13862	-1.4%

TABL E 2 Validation of Storage Usage of TPC-C Compliant Data- bases

All experimental results show that network la- tencies higher than 60 ms, which are typical of most cloud database environments, make the adaptive encryption overhead almost negligible when considering the overall set of operations of the TPC-C benchmark. However, in the ADAPT configuration, for some SQL operations involv- ing the Ope encryption or for the encryption of a high number of parameters through several en- cryption layers (e.g., INSERT), the impact on the response time is visible even for network latencies higher than 120 ms.

If the workload is characterized by many INSERT opera- tions, we can conclude that it is a tenant's duty to solve the tradeoff between ac- cepting adaptive encryption overheads and pay- ing the costs related to an entire database re- en- cryption when workload changes in statically encrypted databases. It is likely that this tradeoff can be solved on the basis of the expected va- riability of the workload. Possible improvements can be achieved by parallel encryption algo- rithms that can leverage multi-threading over different cores, but this research is out of the scope of this paper. On the positive hand, we observe that the presented ADAPT configuration represents a worst case scenario that is fully adaptive, because all database columns are en- crypted with all the onions supported by its data type. On the other hand, the ENC configuration represents a best case scenario that is completely static, because the user manually defines the single encryption scheme to use on each database column.

We observe that a tenant may choose a partially adaptive configuration in which a subset of columns are encrypted with adaptive strategies and others are statically encrypted. As a conse- quence, performance of adaptive encryption for many realistic workloads falls between the ENC and ADAPT scenarios presented in this section.

The validation demonstrates the efficacy of the proposed analytical usage estimation methodol- ogy in the TPC-C workload. Costs evaluations proposed in the following sections are based on the same usage estimations.

V. **Cost Evaluation**

In this section we demonstrate the feasibility of the proposed cost model by applying it to PLAIN, ENC and ADAPT configurations (see Section 5) in real cloud database services. We initially validate the usage estimation method- ology presented in Section 4.3. We then analyze how costs vary for different cloud providers and resource usages. We finally evaluate tenant's costs over a medium-term period equal to three years by considering realistic resource usage increments and cloud price reductions.

5.1 Validation of the Usage Estimation

To validate the usage estimation model, we perform several experiments based on the TPC-C benchmark. First of all, we validate the storage usage estimation model. We deploy nine TPC-C compliant databases of three different sizes: 1, 5 and 10 warehouses (the number of warehouses is the TPC-C parameter that influences the initial database size). For each size, we generate three data- base configurations: PLAIN, ENC and ADAPT. Results are summarized in Table 2. Estimated storage of PLAIN, ENC and ADAPT are calculated by using the analytical model presented in Section 4.3. For each estimated value, we report the estimation error with respect to the measured database size. Errors are ex- pressed as a percentage. We observe that the proposed model always overestimates the data- base size. However, errors show that estimations are close to measured sizes. For PLAIN data- bases, the errors is always below 2%, while for ENC and ADAPT databases the error is always between 5% and 6%. We then validate the net- work usage estimation model. We deploy PLAIN, ENC and ADAPT TPC-C compliant databases, each having 10 warehouses. We ob- serve that network consumption is invariant with respect to the number of warehouses, because it only depends on encryption and query workload. We measure the network usage of the PLAIN database, and we obtain an average of 7,162 Bytes per transaction. By using Equation (8). we estimate np ¼ k 548. Hence, we determinek ¼ 13:07. Then we use this value of k to deter- mine the estimated network usage of ENC and ADAPT configurations. We compare these val- ues with the experimentally measured

network usages. Results are summarized in Table 3. Es- timations are quite accurate, since we achieve

VI. **Conclusions**

There are two main tenant concerns that may prevent the adoption of the cloud as the fifth utility: data confidentiality and costs. This paper addresses both issues in the case of cloud data- base services. These applications have not yet received adequate attention by the academic li- terature, 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 interme- diate 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 ex- periments 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 encryp- errors of 1:2% and 1:4% for the ENC and tion. Moreover, we propose a model and a me- ADAPTconfigurations,respectively. The valida- tion demonstrates the efficacy of the proposed analytical usage estimation methodology in the TPC-C workload. Costs evaluations proposed in the following sec- tions are based on the same usage estimations.

6.2 Analysis of Cloud Database Costs

We analyze cloud database costs with respect to different cloud provider offers and different sto- rage and network usages. We consider a billing period equal to one month, and 24/7 availability (730 uptime hours per month). We initially es-timate the monthly costs of a cloud data- base service in the PLAIN, ENC and ADAPT configurations with respect to a plaintext storage usage of 100 GB and a plaintext network usage of 100 GB. In Table 4, we report the results for the following cloud instances: Small, Large, and High Memory: Double Extra Large from Ama- zon RDS Premium P1 and Premium P2 from SQL Azure.

Methodology that allow a tenant to estimate the costs of plain and encrypted cloud database ser- vices 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.

VIII. 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 Genera- tion Comput. Syst., vol. 25, no. 6, pp. 599–616, 2009.

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

[3] H.-L. Truong and S. Dustdar, “Composable cost estimation and monitoring for computational appli- cations in cloud computing environments,” Procedia Comput. Sci., vol. 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 Proc. ACM/IEEE Conf. Supercomputing, 2008, pp. 1–12.

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

[6] G. Wang, Q. Liu, and J. Wu, “Hierarchical attribute-based encryp- tion for fine-grained access control in cloud storage services,” in Proc. 17th ACM Conf. Comput. Commun. Security, 2010, pp. 735–737.

[7] Google. (2014, Mar.). Google Cloud Platform Storage with server- side encryption [Online]. Available: http://googlecloudplatform. blogs- pot.it/2013/08/google-cloud-storage-now-provides.h tml.

[8] L. Ferretti, M. Colajanni, and M. Marchetti, “Distributed, concurrent, and independent access to encrypted cloud databases,” IEEE Trans. Parallel and Distributed Systems, vol. 25, no. 2, Feb. 2014.

[9] R. A. Popa, C. M. S. Redfield, N. Zeldovich, and H. Balakrishnan, “CryptDB: protecting confidentiality with encrypted query processing,” in Proc. 23rd ACM Symp. Operating Systems Principles,Oct. 2011.

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

[11] A. Boldyreva, N. Chenette, and A. O’Neill, “Order-preserving encryption revisited : Improved security analysis and alternative solutions,” in Proc. Advances in Cryptology – CRYPTO 2011. Springer, Aug. 2011.

[12] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in Proc. Advances in Cryptology – EURO-CRYPT99. Springer, May 1999.

[13] D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. IEEE Symposium on Security and Privacy., May 2000.

[14] L. Ferretti, F. Pierazzi, M. Colajanni, and M. Marchetti, "Security and confidentiality solutions for public cloud database services," in Proc. Seventh Int'l Conf. Emerging Security Information, Systems and Technologies, Aug. 2013.

Author Profiles:

MD RafiYuddin, Pursuing his M. Tech from KIET, Kakinada Institute of Engineering & Technology, Korangi, E.G.dt ,JNTUK, AP, India,. He Received his B. Tech From Godavari Institute Of Engineering & Technology, Rajahmundry, E.G.dt,JNTUK, AP, India,. His Areas of interest is cloud computing.

Mr .Sheik Ahmad Shah is working as Asst.professor in KIET .He has 8Years of Teaching Experience .He completed his B.Tech in 2007.He completed his M.Tech(IT) from JNTU kakinada in 2010.His Areas of interests includes Networking & security.