International Journal of Engineering Research and General Science Volume 2, Issue 6, October-November, 2014
ISSN 2091-2730

ANALYSIS OF EFFECTIVE MULTI USER DISTRIBUTION KEY MANAGEMENT SCHEME IN CLOUD DATABASE

Dr.P.SUMITRA, M.Sc., M.Phil.,MCA.,Ph.D.,[1] M.KAVINNELA,[2]
ASSISTANT PROFESSOR, M.PHIL FULL-TIME RESEARCH SCHOLAR,
sumitravaradharajan@gmail.com Kavi92.msc@gmail.com

DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS,
VIVEKANANDHA COLLEGE OF ARTS AND SCIENCES FOR WOMEN, TAMILNADU, INDIA.

Abstract: Database is a service paradigm poses several research challenges in terms of security and cost evaluation from a tenant‘s point of view. The cloud database as a service is a novel paradigm that can support several Internet-based applications, but its adoption requires the solution of information confidentiality problems. 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 Database, Adaptive Encryption, SQL operations, Metadata, Distributed SQL operation, Multi key Distribution

I. INTRODUCTION

Managing and providing computational resources to client applications is one of the main challenges for the high performance computing community framework. To monitoring resources existing solutions rely on a job abstraction for resource control, where users submit their applications as batch jobs to a resource management system responsible for job scheduling and resource allocation [1]. This usage model has served the requirements of a large number of users and the execution of numerous scientific applications. However, this usage model requires the user to know very well the environment on which the application will execute. In addition, users can sometimes require administrative privileges over the resources to customize the execution environment job model. The manage and increasing availability of virtual machine technologies has enabled another form of resource control based on the abstraction of containers. A virtual machine can be leased and used as a container for deploying applications [2]. Under this scenario, users lease a number of virtual machines with the operating system of their choice; these virtual machines are further customized to provide the software stack required to execute user applications. This form of resource control has allowed leasing abstractions that enable a number of usage models, including that of batch job scheduling [3].

Investigate whether an infrastructure base operating its local cluster can benefit from using Cloud providers to improve the performance of its users' requests. The evaluate scheduling strategies suitable for a distributed cloud that is managed by proposed technology to improve its SQL operation with adaptive encryption data values. These strategies aim to utilize remote resources from the Cloud to augment the capacity of the SQL operation. However, as the use of Cloud resources incurs a cost, the problem is tothe price at which this performance improvement is achieved. The aim to explore the trade between performance improvement and cost. The decryption and encrption key. As an application, they suggested private data banks: a user can store its data on an untrusted server in encrypted form, yet still allow the server to process, and respond to, the user‘s data queries (with responses more concise than the trivial solution: the server just sends all of the encrypted data back to the user to process). Since then, cryptographers have accumulated a list of ―killer‖ applications for fully homomorphism encryption. However, prior to this proposal, we did not have a

viable construction.

The rest of this paper is organized as follows. Section 2 presents the related work followed by the main contribution distributed cloud as well as the problem definition in Section 3. Section 4 gives a brief introduction to multi key distribution while and explains the proposed approach. Finally, Section 5 presents the evaluation of the algorithm followed by the conclusions described in Section 6.

II. RELATED WORK

An effective the privacy of information stored in cloud databases represents an important 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. Although data encryption seems the most intuitive solution for privacy, its application to cloud database services is not trivial, 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 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 database, 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 overheads, 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 [6].

This paper has a focus on database services and takes an opposite direction by analysis the cloud service costs from a boarder‘s point of generation. This approach is rather original because relate work is evaluate the process and connection of porting logical applications to a distributed cloud platform, such as [4] focusing on specific astronomy software and a specific distributed cloud provider and [5] [9] presenting a compassable cost estimation model for some classes of logical applications. Besides the focus on a different context (logical versus database applications), the proposed model can be applied to any distributed cloud database service provider, and it takes into account that over a medium-term period the database workload and the distributed cloud prices may vary.

III. MAIN CONTRIBUTIONS

In the existing system, all data and metadata stored in the cloud database are encrypted and application running is 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 is not encrypted, whereas information is always encrypted before sending it to the cloud database. When an application issues a new SQL operation, the encrypted database interface contacts the encryption engine that retrieves the encrypted metadata and decrypts them with the master key. To improve performance, the plain metadata are cached locally by the client. After obtaining the metadata, 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 encrypted database interface

- Multi-user key distribution scheme is not proposed to provide data to the same group of users
  - Encryption cost and thereby data transmission cost is more.
  - Same kind of encryption is maintained for all the data saved in the cloud nodes.

IV. PROPOSED PROTOCOL

Like existing system, proposed system also manages the data using both cloud server side and client side. In addition, user group is maintained so that a single key is distributed to multiple users in the same group to reduce the key preparation overhead for each user. This makes less computation overhead in both client and server side. Also, based on the security level, different data is

encrypted with different encryption mechanism and allowed to secure the data in inexpensive manner.

- Multi-user key distribution scheme is proposed to provide data to the same group of users.
- Encryption cost and thereby data transmission cost is less.
- Different kind of encryption is maintained for various data saved in the cloud nodes based on the security level requirement.

ALGORITHM WORK MODEL

1. Records Collection

In this step, the records to be saved in cloud database (for example, employee and their attendance details) are keyed in and saved. Employees and Attendance table are used to save the records. The plain meta data keyword(s) are also obtained for the given record and saved along with the employee data.

2. Records Encryption

In this module, the using Triple Data Encryption Standard and Advanced Encryption Standard algorithm, the records (employee details) is encrypted and saved in cloud database. EncEmployees and EncAttendance table are used to save the records. Meta data keywords are also encrypted and saved in the database. So the cloud database contains both _encrypted meta 'data' and _encrypted data'.

3. User

In this first step, the user id, username and password along with email id details are added and saved into _'Users'table. In the second step, the user group id, user group name details are added and saved into _'UserGroup'table. In the third step, the user group id, user id details are fetched from _'User Group' and _'Users' table and saved into _'Users Assigned' table.

4. Distribute Key to User Group

In the step, the user group ids are fetched from _' User Group' table and encryption master keys which are used to encrypt the employees and attendance details are given to that user group. Mails are sent to all the users in the user group with encryption master keys.

5. Client Application

In this step, application is running on a legitimate client. An SQL operations (e.g., SELECT, INSERT, UPDATE and DELETE) is to encrypted cloud database through the encrypted database interface. Data transferred between the user application and the encryption engine is not encrypted, whereas information is always encrypted before sending it to the cloud database. When a user issues a new SQL operation, the encrypted database interface contacts the encryption engine that retrieves the encrypted metadata and decrypts them with the given master key. After obtaining the metadata, 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 encrypted database interface.

V. EXPERIMENTAL RESULTS

Figure 1.1 is describing existing system data transmission cost analysis. In this figure contains number of data size in SQL operation and average encryption data size in SQL operation are shown below

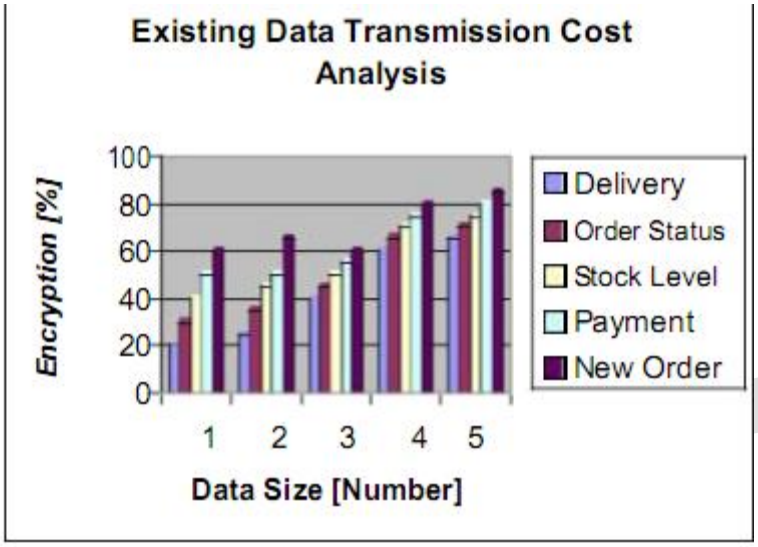

Figure 1.1 Existing Data Transmission Cost Analysis

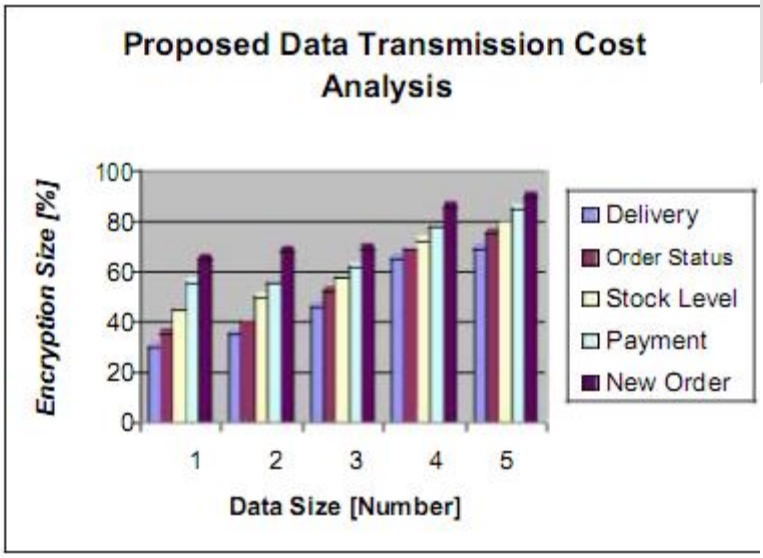

Figure 1.2 Proposed Data Transmission Cost Analysis

Figure 1.2 is describing existing system data transmission cost analysis. In this figure contains number of data size in SQL operation and average encryption data size in SQL operation are shown below. The comparison for existing and proposed system data transfer cost analysis is better than the proposed multi key distribution model for distributed could environments.

ACKNOWLEDGMENT

I express my deep gratitude and sincere thanks to my supervisor **Dr.P.Sumitra,M.Sc.,M.Phil.,MCA.,Ph.D., Assistant Professor, Department of Computer Science** at **Vivekananda college of Arts and Sciences for Women** for her valuable,

Suggestion, innovative ideas, constructive, criticisms and inspiring guidance had enabled me to complete the paper present work successfully

VI.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. 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 based. By applying the model to actual cloud provider cost, we can determine the encryption and adaptive encryption costs for data privacy. Future research could analysis the proposed model for distribution user key schemes and under different threat model hypotheses

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. Syst., vol. 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.
3. Sebastopol, CA, USA: O‘ReillyMedia, Inc., 2009.
4. H.-L. Truong and S. Dustdar, ―Composable cost estimation and monitoring for computational applications in cloud computing environments,‖ Procedia Comput. Sci., vol. 1, no. 1, pp. 2175–2184, 2010.
5. E. Deelman, G. Singh, M. Livny, B. Berriman, and J. Good, ―The cost of doing science on the cloud: The montage example,‖ in
6. Proc. ACM/IEEE Conf. Supercomputing, 2008, pp. 1–12.
7. H. Hacig€ um€ u¸ s, B. Iyer, and S. Mehrotra, ―Providing database as a service,‖ in Proc. 18th IEEE Int. Conf. Data
8. Eng.,Feb.2002, pp. 29–38.
9. G.Wang, Q. Liu, and J. Wu, ―Hierarchical attribute-based encryption for fine-grained access control in cloud storage services,‖ in
10. Proc. 17th ACMConf. Comput. Commun. Security, 2010, pp. 735–737.
11. Google. (2014, Mar.). Google Cloud Platform Storage with server side encryption [Online]. Available: HTTP://GOOGLECLOUDPLATFORM. blogspot.it/2013/08/google-cloud-storage-now-provides.html.
12. H.Hacig, u¸ s, B. Iyer, C. Li, and S.Mehrotra, ―Executing SQL over encrypted data in the database-service-provider model,‖ in Proc. ACMSIGMODInt‘lConf.Manage.Data, Jun. 2002,pp. 216–227.
13. L. Ferretti, M. Colajanni, and M. Marchetti, ―Distributed, concur- rent, and independent access to encrypted cloud databases,‖
14. IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 2, pp. 437–446, Feb. 2014.