IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

Load Balancing Model for Performance, Accuracy, and Precision for Secure Cloud Transactions

M. Naga Divya
PG Student, Shri Vishnu Engineering College for Women, Bhimavaram
Computer science and engineering

Y. Ramu,
Associate Professor in CSE,
Shri Vishnu Engineering College for Women, Bhimavaram

Abstract

Technology and its service it's for the best of the agility, edge and other related stuffs like which we call as the generation for speed, accuracy and secure for the data. Data which is the most important in terms of performance where we consider the speed of data transfer and accuracy; it's may be 99.000001 but cannot be expected 100%, which is to be improved. In this paper, we followed an approach with more emphasis on the performance, accuracy, and robustness of the data in terms of the transaction, which involves several technical level tasks. At first, moving through the network and making it secured through the cryptography and next to focus on the performance with respective to parallel distributed mechanism and the high end protocol of the dedicated bandwidth in order to avoid the network collision. In this work, we have overcome the public protection issues to provide privacy in terms of public audit, and data in cryptographic model. Experiments are carried out on real time datasets.

Index Terms—load balancing model; public cloud; cloud partition

1. Introduction

The web has made it easy to provide and consume content of any form. Building a web page, starting a blog, and making them both searchable for the public have become a commodity. Nonetheless, providing an own web application/web service still requires a lot of effort.[9] One of the most crucial problems is the cost to operate a service with ideally availability and acceptable latency. In order to run a large-scale service like YouTube, several data centers around the world are needed. [4] Running a service becomes particularly challenging and expensive if the service is successful: Success on the web can kill! In order to overcome these issues, utility computing (a.k.a., cloud computing) has been proposed as a new way to operate

IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

services on the internet. Some techniques tradeoff between the consistency and response times of a write request. The authors in develop models for transactional databases with eventual consistency, in which an updated data item becomes consistent eventually. [7]

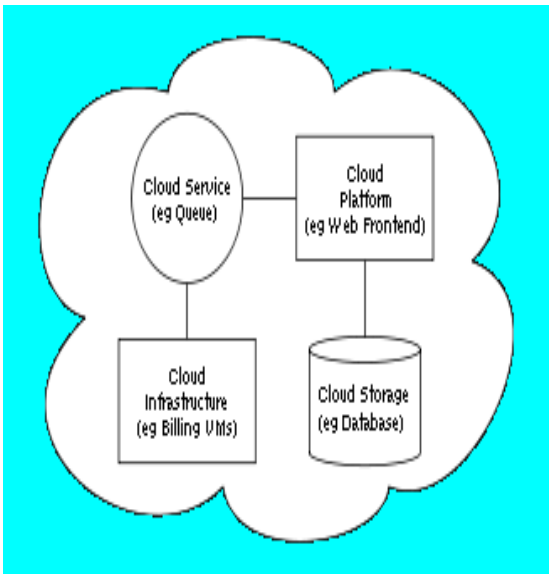


Fig.1.1. Illustration of the Different Cloud Service

Other approaches use data versioning to keep a reasonable amount of delay. However, data versioning and compromised consistency are not favorable for transactional databases. Some market available cloud-based databases use the quorum-based protocol to maintain consistency, while other cloud based relational database solutions use pull-based consistency techniques. [9]

2. Related Work

Although the advantages for building applications in the cloud are compelling, they come with certain limitations. By today, there exists no consensus on cloud services. Thus, different providers offer different functionality and interfaces, which makes it hard to port applications from one provider to another. Furthermore, the systems sacrifice functionality and consistency to allow for better scaling and availability.[5]If more functionality and/or consistency is required it has to be built on top. Although some cloud providers offer best-practice guidelines on building applications in the cloud, the new trade-offs - especially for applications which may require stronger consistency guarantees (such as database applications) - are not addressed at all. Auto-scaling is one of the advantages provided by a cloud computing platform. [6]

IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

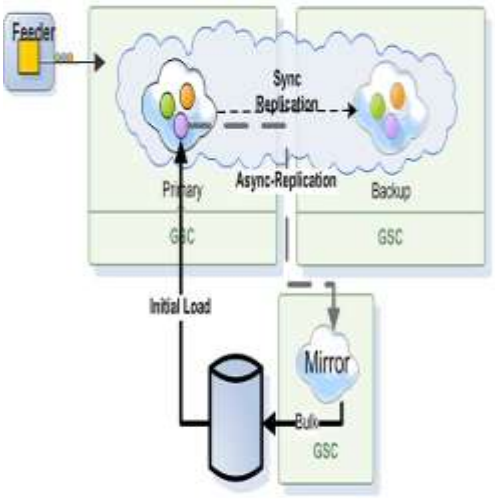


Fig.2.1. Synchronization of the Data with RAD in the cloud Structure

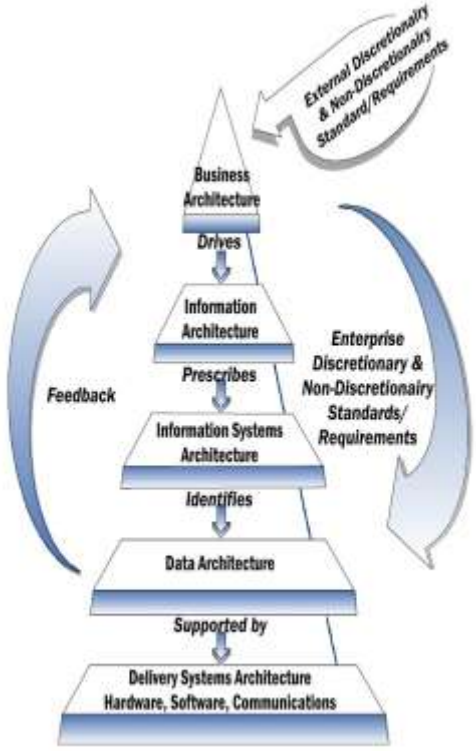
Each and every application deployed on a cloud platform should be able to take advantage of this feature. Decisions for scaling-up or scaling-down have a significant impact on performance and resource usage, because there is an overhead associated with the auto-scaling process. It is very important to distinguish between the actual change in the workload and an anomaly.[7] It seems that this problem can be solved by utilizing the recently proposed LT network codes (LTNC) which provides efficient decoding at the cost of slightly more communication in the single-source broadcasting scenario. However, after several rounds of repair with same recoding operations regulated in LT network codes, data users experience decoding failure with high probability. There are various types of searchable encryption methods, such as Symmetric Searchable Encryption (SSE) and Asymmetric Searchable Encryption (ASE). Each one of these methods is suitable for particular application scenarios. For instance, Symmetric Searchable Encryption (SSE) is useful in such cases where the customer or user who requests (searches) the data is also the one who creates it initially. The customer needs to send a token that contains a keyword w and the provider will return the document (encrypted) that contains w. Furthermore, the cloud provider will know nothing about the data unless he/she knows the token.

3. Methodology

Cloud computing shares the basic theme of previous paradigms associated with the provisioning of computing infrastructure. [13] However, cloud computing differs in that it shifts the location of the infrastructure to the network to provide basic components. These basic components, such as storage, CPUs, and network bandwidth, are provided as a service by specialized service providers at a low unit cost. Users of these services are guaranteed not to be worried about scalability and backups because the available resources are virtually infinite and failed components are replaced without any service interruption and data loss. Transactional data management is the heart of the database industry. A transaction is a logical unit of works that consists of a series of read and/or writes operations to the

IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

database. Nowadays, almost all business transactions are conducted through transactional data management applications.



3.1. Flow Diagram of the Proposed System

Technology has its own significance at the time when people having the extension for the more and more research and it's from Abacus to today's' cloud Computing.[10] In the context of revolution of technology and its great advantage to its social, behavioral and other technical aspect where we come across the best of the cloud to province the virtual global village as the global world. In order for an attack against Anonymous Cloud to succeed, the manager or master node (or both) must be malicious. Managers are the only principals that receive decrypt able access tokens or credentials, and all other communications involving pseudonyms and data are conducted via Tor circuits having the master node as the only un-trusted endpoint. Managers are separate from CPs and have a much smaller attack surface because they do not process customer-submitted computations.[4] Our experiments therefore assume that managers are trusted, but that master nodes are always malicious. These applications typically rely on guaranteeing the ACID (Atomicity, Consistency, Isolation and Durability) properties provided by a database and they are fairly write-intensive. Performing a write operation could be time consuming due to ACID property maintenance, especially for a system distributed over different geographic locations like a cloud.

IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

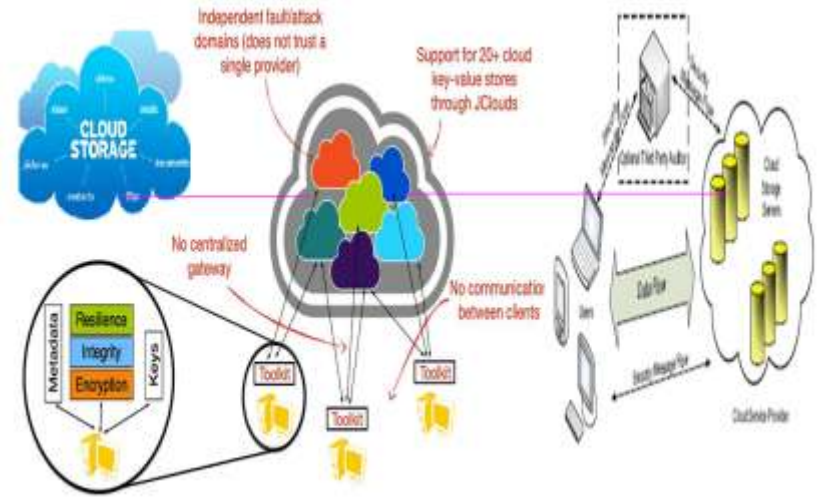


Fig.3.2. Architecture of the secured Transaction in the Cloud Tired Structure

Analytical data management systems can tolerate such time delays, but for transactional data management systems, such time delays are quite unacceptable. A consistent database must remain consistent after the execution of each and every transaction. Any kind of inconsistent state of the data can lead to significant damage, which is completely unacceptable. Service availability and data reliability are usually achieved by creating a certain number of replicas of the data distributed over different geographical areas. Amazon's S3 cloud storage service replicates data across 'regions' and 'availability' zones so that data and applications can persist even in an entire location black out. Maintaining consistency in such a distributed platform is time consuming and could be expensive in terms of and concurrency control is major issues in the execution of a transaction in any computing platform.

Concurrency control ensures that when database transactions are executed concurrently, the results of the transactions are consistent and, meaning they are the same as if they were executed serially. Consistency maintenance becomes more complex by allowing concurrent access to a database, especially in a distributed platform like a cloud platform. Several strategies have been proposed to implement a transaction manager in a cloud platform, but each has their limitations.[8] The strategies are to divide the transaction manager into multiple levels to perform both distributed and local affairs in transaction execution in a cloud platform. Specifically, the strategy is designed for web-based transactions, but data can become unavailable during failures,

IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

while the proposed strategy in can support only a subset of database operations. For the strategy in, transactional logs are stored in a file instead of executed on the actual database, but this strategy is applicable to applications that have only a few updates to the data. In this context, we have taken In-Memory Processing in order to optimize 100% than the existing one.

3.1 Evaluation and Analysis and results

In recent years, the number of research communities focusing on large volume data sets has increased considerably to include such traditional enterprise applications as Web search, astronomy, biology, earth science, digital entertainment, natural-language processing and social network analysis.[5] The ubiquity of huge data sets is increasing the number of users as well as the number of developers of data management technologies, and will require new ways of thinking in the database research field.

Dataset: Twitter trend Dataset

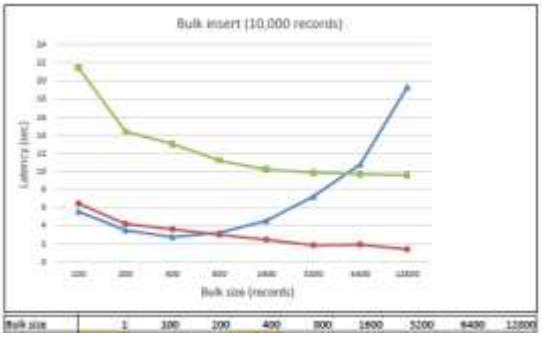


Fig.3.1.1. Variation of the Dataset of the Twitter Trend w.r.t. Existing Load balancer

As a result, few mainstream mobile computing devices have adopted these technologies. Moreover, many of these solutions rely on software obfuscation, which does not provide rigorous guarantees, since clever attackers can potentially reverse the obfuscation the sharp increase in communications overhead. One approach is to provide Database as a Service (DaaS) in the cloud. DaaS is another specialized version of SaaS which is specifically developed for database management in clouds. In the Dataset of the Twitter trend of 1 Million record, we have taken in to case study gives the glimpse as below.

4. Conclusion and Future work

Consistency Rationing can significantly lower the overall cost and improve the performance of cloud-based database systems. Our experiments show further that adapting the consistency by means of temporal statistics has the biggest potential to lower the overall cost while maintaining acceptable performance. Future work includes in particular extensions to the notion of probabilistic consistency guarantees. Possible improvements include: better and faster statistical methods, automatic optimizations with regards to other parameters (e.g., energy consumption),

adding budget restrictions to the cost function, and relaxing other principles of the ACID paradigm (e.g., durability). Further, the notion of probabilistic consistency guarantee might also be applicable to deal with the CAP theorem, if availability instead of cost is the main optimization factor. However, going into more detail is beyond the scope. In the Future, adoption of user-centric security models and shifting certain parts of communication and computation to the client side allows us to provide the cloud consumers with more visibility and control over their resources.

Reference

[1] M. Armbrust *et al.*, "Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep., Feb. 2009.

[2] S. Das, D. Agrawal, and A. El Abbadi, "Elastras: an elastic transactional data store in the cloud," in *USENIX HotCloud*, 2009.

[3] D. J. Abadi, "Data management in the cloud: Limitations and opportunities," IEEE Data Engineering Bulletin, Mar. 2009.

[4] A. J. Lee and M. Winslett, "Safety and consistency in policy-based authorization systems," in *ACM CCS*, 2006.

[5] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams, "X.509 internet public key infrastructure online certificate status protocol - ocsp," RFC 2560, Jun. 1999, http://tools.ietf.org/html/rfc5280.

[6] E. Rissanen, "extensible access control markup language (xacml) version 3.0," Jan. 2013, http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html.

[7] D. Cooper *et al.*, "Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile," RFC 5280, May 2008, http://tools.ietf.org/html/rfc5280.

[8] J. Li, N. Li, and W. H. Winsborough, "Automated trust negotiation using cryptographic credentials," in *ACM CCS*, Nov. 2005.

[9] L. Bauer *et al.*, "Distributed proving in access-control systems," in *Proc. of the IEEE Symposium on Security and Privacy*, May 2005.

[10] J. Li and N. Li, "OACerts: Oblivious attribute based certificates," *IEEE TDSC*, Oct. 2006.

[11] J. Camenisch and A. Lysyanskaya, "An e_cient system for nontransferable anonymous credentials with optional anonymity revocation," in *EUROCRYPT*, 2001.

[12] P. K. Chrysanthis, G. Samaras, and Y. J. Al-Houmaily, "Recovery and performance of atomic commit processing in distributed

IJMTARC – VOLUME – IV – ISSUE - 16 - DEC 2016 ISSN: 2320-1363

database systems," in *Recovery Mechanisms in Database Systems*. PHPTR, 1998.

[13] M. K. Iskander, D. W. Wilkinson, A. J. Lee, and P. K.Chrysanthis, "Enforcing policy and data consistency of cloud transactions," in *IEEE ICDCS-SPCC*, 2011.

[14] G. DeCandia *et al.*, "Dynamo: amazons highly available key-value store," in *ACM SOSP*, 2007.

[15] F. Chang, *et al.*, "Bigtable: A distributed storage system for structured data," in *USENIX OSDI*, 2006.