Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 5, May 2015, pg.901 – 904

RESEARCH ARTICLE

Two Level Auditing Structure: Consistency as a Service

Shambhavi
Department of Computer Science & Engineering, VTU PG Centre, Kalburagi, India
tanvi.maski002@gmail.com

Asst. Prof. Ranjana B Nadagoudar
Department of Computer Science & Engineering, VTU PG Centre, Kalburagi, India
ranjanapriya@gmail.com

Abstract-As we know Cloud storage services are providing very large amount of advantages. Because of this cloud services are becoming very popular most of the all popular companies are using all cloud service as always on access. . To provide this ubiquitous always on service most of the cloud service provider (CSP) maintains each piece of data on geographically distributed servers. But the main problem with this technique is that, it is very expensive and some to fail to provide required consistency of service. To overcome this problem, we propose to use a new approach of service (i.e. Consistency as a Service (CaaS)) this paper, firstly concentrate on a consistency as a service (CaaS) model, which has a large data cloud and multiple small audit clouds. In the CaaS model, a data cloud is formed by a CSP, and a group of users form an audit cloud that can verify whether the data cloud provides the promised level of consistency i.e. quality of service or not, to prove this we are proposing two-level auditing strategy which require loosely synchronized clock for ordering operations in an audit cloud. Then perform global auditing by global trace of operations through randomly electing an auditor from an audit cloud. And then proposing log auditing for tracing all operations.

Keywords— Cloud Storage Systems, consistency as a service (CaaS), two-level auditing, log auditing.

I. INTRODUCTION

Clouds computing is become more popular as it provides guaranteed services like data storage, virtualized infrastructure etc. e.g. Amazon, Simple DB etc. By using the cloud services, the customers or user can access data stored in a cloud anytime and at anywhere using any device, and customer ensure about less capital investment. To provide promised always on 24/7 access, the cloud service provider(CSP) stores data replicas on multiple geographically distributed servers. The main drawback of using the replication technique is it is very expensive to achieve strong consistency, and user is ensured to see the latest updates. Many CSPs (e.g., Amazon S3) provide only eventual i.e. updates are visible definitely but not immediately. E.g. Domain name system (DNS), but the eventual consistency is not interesting for all applications and which require strong consistency. Some applications like social networking sites require causal i.e. strong consistency. Thus the different applications require different level of consistency. We propose novel consistency as a service (CAAS) model. The CaaS model consists of, A large data cloud formed by CSP and multiple audit clouds formed by group of users worked on project or document that can check whether the data cloud provide promised level of consistency or not. Two-level auditing structure which require only a

loosely synchronized clock for ordering operation in an audit cloud then perform global auditing with a global trace of operations periodically an auditor is elected from an audit cloud. Local auditing is concentrate on monotonic-read and read-your write consistencies, which can be performed by two level auditing structure algorithms while Global auditing focuses on causal consistency, in which construct a directed graph. If the constructed graph is a directed acyclic graph also called as precedence graph, we claim that causal consistency is preserved. We determine the tracking of operation by log auditing.

II. RELATED WORK

This section reviews some of the related work followed by discussing their connections and differences with the proposed approach.

A. TANENBAUM AND M. VAN STEEN [2] proposes the two classes of consistency models i.e. data centric consistency and client centric consistency. Data centric consistency focuses on internal state of storage system while the client centric consistency focuses on what specific customer want. To verify these levels of consistency provided by CSP is done by two ways: trace based verification and benchmark based verification.

W. VOGELS [3] sates that strict consistency is never required in practice also it is considered to be harmful. Thus The Amazon.com technology provides a set of more advanced business and infrastructure services that are implemented using scalable distributed systems. In this environment we can analyse a number of particular data access patterns, each with their own consistency requirements. Thus to provide a collection of more diverse business processes the different patterns are provided.

BREWER [4] AND PUSHING THE CAP [5] states that Modern distributed Systems have adopted new types of data store that are not secure to provide strong Consistency. Thus the CAP theorem and its evolution will influence on these systems which provide scarifies strict consistency andprovide weak consistency with high availability .

E. ANDERSON, X. LI, M. SHAH, J. TUCEK, AND J. WYLIE[6]propose efficient algorithm that analyzing the trace of interactive operations between the client machines and key value store i.e. data store to report weather the trace is safe ,regular or atomic, because number of data store provide platform for always on globally distributed applications. To meet their goals they scarify strong consistency and provide high availability. While some of the tools used to verify the violations. And conclude whether system is good or bad and provide promised level of consistency or not.

W. GOLAB, X. LI, AND M. SHAH [7] provides an online verification algorithms by using GK algorithms for several known consistency properties. Also we consider how to determine the severity of the violations, if consistency violations? And this is check by two matrices: value of old reads, and another is the commonality of violations. For not fresh data again consider time-based staleness and operation-count-based staleness. Finally present an efficient algorithm that calculates these quantities.

T. KRASKA, M. HENTSCHEL, G. ALONSO, AND D. KOSSMANN [8] Cloud storage solutions promise high scalability and low cost. Existing solutions, however, differ in the degree of consistency they provide. Our experience using such systems indicates that there is a non-trivial trade-off between cost, consistency and availability. High consistency implies high cost per transaction and, in some situations, reduced availability. Low consistency is cheaper but it might result in higher operational cost because of, e.g., overselling of products in a Web shop. A new transaction paradigm that not only allows designers to define the consistency guarantees on the data instead at the transaction level, but also allows to automatically switch consistency guarantees at runtime.A new concept called Consistency Rationing to optimize the runtime cost of adatabase system in the cloud when inconsistencies incur a penalty cost. The optimization is based on allowing the database to exhibit inconsistencies if it helps to reduce the cost of a transaction but does not cause higher penalty costs.Categories: A, B, and C. the A category ensures strong consistency guarantees and shows high cost per transaction. The C category ensures session consistency, shows low cost, but will result in inconsistencies. Data in the B category is handled with either strong or session consistency depending on the specified policy.

III. IMPLEMENTATION

This section consist of three models i. e. consistency as a service (CaaS) model, user operation table (UOT) with which each user records his operations and two-level auditing structure.

3.1 Consistency as a Service (CAAS) Model

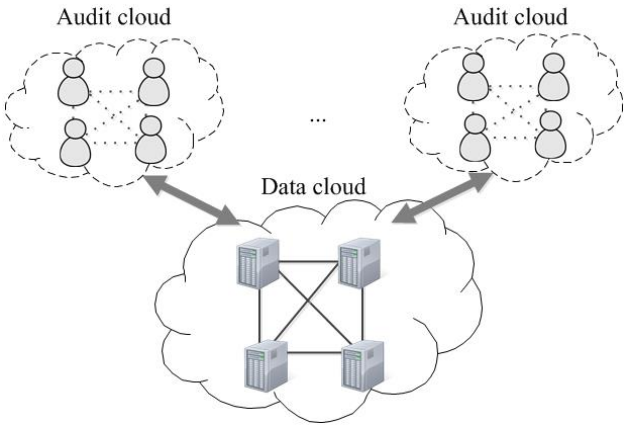

Fig 1: Caas model

An audit cloud consists of a group of users that work together on a job, e.g., a document or a program. We consider that each user in the audit cloud is identified by unique ID. Before assigning job to the data cloud, an audit cloud and the data cloud will engage with a service level agreement (SLA), which demands the promised level of consistency should be provided by the data cloud.

The audit cloud exists to verify whether the data cloud violates the SLA or not..

3.2 User Operation Table (UOT)

Each user maintains his own User Operation Table (UOT)for recording his trace of operations. Each record in the UOT is described by elements like Operation, logical vector, and physical vector. While issuing an operation, a user from an audit cloud will record his operation in UOT, as well as his current logical vector and physical vector. Each user will maintain a logical vector and a physical vector to track the logical and physical time when an operation happens ,respectively.

3.3 Two-level auditing structure algorithms

Local Auditing: Each user independently performs local auditing with his UOT with two consistencies; Monotonic read consistency, which requires that a user must read either a new value or same value Read-yours-write consistency, which require a user, always read his latest update.

Local consistency auditing - For auditing local consistency - attackers while accessing file and checking read or write

S1- Check the user request

S2 - Check the consistency type(Read or Write)

S3 -Allow the file accessing based on read or write users

S4- Capture the attackers if the rule is violated

Global Auditing: Global auditing is performed by global trace of operations of all users operations with their following consistency

Global Consistency auditing - Data integrity and auditing and checking all the clouds for accessing file

S1- Check the number of users file request

S2 - Check file in all cloud(here cloud called as edges) and register all cloud tracing in audit cloud

S3 -Allow the file if file is not integrated

S4- Capture the file auditing if the file is modified

3.4 Log auditing

Here this log auditing mainly used for tracking purpose .that is when ever cloud servers and users logged in to perform any access .then it will record everything as at what time they used and what they did as weather they modified the data or simply saw the information everything is recorded and this information will be sent to the user email address. And administrator can view all the information which mainly provides security for the system.

IV. RESULT AND DISCUSSION

Here we are implemented consistency as a service model to verify whether they provide promised level of consistency and implemented two level auditing structure to perform audit operations as local and globally auditing and for tracking the information we implemented log auditing which provides security.

V. CONCLUSION

In this paper, we argued that strong consistency requirements should be adopted only for data objects crucial for application correctness, otherwise weaker forms of data consistency should be adopted. We presented a consistency as a service (CaaS) model and a two-level auditing structure that helps users to verify whether the cloud service provider (CSP) is providing the promised consistency, and performing local and global audit operation .With the CaaS model, the users can assess the quality of cloud services and choose a right CSP among various candidates, e.g., the least expensive one that still provides adequate consistency for the users' applications and with help of log auditing we can provide security by tracking.

ACKNOWLEDGEMENT

I would like to express my sincere thanks to my guide Asst Prof. Ranjana.B. N for her motivation and useful suggestions which truly helped me in improving the quality of this paper. I take this opportunity to express my thanks to my teacher, family and friends for their encouragement and support.

REFERENCES

[1] in Liu, Guojun Wang, IEEE, Member, and Jie Wu, IEEE Fellow "Consistency as a service: Auditing cloud consistency", IEEE Vol 11 No.1, July 2014.

[2] A. TANENBAUM AND M. VAN STEEN, Distributed Systems: Principles andParadigms. Prentice Hall PTR, 2002

[3] W. VOGELS, "Data access patterns in the Amazon.com technology platform," in Proc. 2007 VLDB.

[4] E. BREWER, "Towards robust distributed systems," in Proc. 2000 ACM PODC.

[5] Pushing the CAP: strategies for consistency and availability,"Computer, vol. 45, no.2, 2012

[6] E. ANDERSON, X. LI, M. SHAH, J. TUCEK, AND J. WYLIE,"What consistency does your key-value store actually provide," in Proc. 2010 USENIX HotDep.

[7] W. GOLAB, X. LI, AND M. SHAH, "Analyzing consistency properties for fun and profit," in Proc. 2011 ACM PODC.

[8]. T. Kraska, M. H[entschel, G. Alonso, and D. Kossmann, "Consistency rationing in the cloud: pay only when it matters," in Proc. 2009

[9]Yogesh Chimate et al," Review on preserving caas mode"l, Vol. 5 (6)

