ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3, Special Issue 1, April 2015

An Efficient and Secure System for Cloud Transactions using Policy Based Access Control Mechanism

Lingaraja A J

M.Tech, CSE, Alva's Institute of Engineering and Technology, Moodbidri, D. K, Karnataka, India

Abstract: Consider a distributed file storage system having several servers and storages in different locations deployed over the cloud. A file can be uploaded to the nearest server where an authorized user is located and other authorized users can download the file from the same location. When a user tries to access the file from a different location, the nearest server has to fetch the latest version of the file from the original server, save it in its local storage and allow the requested user to access that file. But the same file is now stored in more than one place and hence the consistency problem arises. In this paper, this issue is addressed and resolved by the use of authorization policies to protect data from unauthorized access and an auditing strategy that checks whether the file is consistent among all the servers and ensures that the user gets the latest version of the file. Third party auditor is used for this purpose. Apart from consistency, the system also provides an attribute based access control to provide better security with policy management system.

Keywords: Distributed file storage, cloud databases, authorization policies, consistency, third party auditing.

I. INTRODUCTION

Cloud computing has recently emerged as a computing paradigm in which storage and computation can be outsourced from organizations to next generation data centers hosted by companies such as Amazon, Google, Yahoo, and Microsoft. Such companies help free organizations from requiring expensive infrastructure and expertise in-house, and instead make use of the cloud providers to maintain, support, and broker access to high-end resources. One of the most appealing aspects of cloud computing is its elasticity, which provides an illusion of infinite, on-demand resources [1] making it an attractive environment for highly scalable, multi tiered applications. Despite the efforts of key-value stores like Amazon's Simple DB, Dynamo, and Google's Big table to provide scalable access to huge amounts of data, transactional guarantees remain a bottleneck [2].

To provide scalability and elasticity, cloud services often make heavy use of replication to ensure consistent performance and availability. As a result, many cloud services rely on the notion of eventual consistency when propagating data throughout the system. This consistency model is a variant of weak consistency that allows data to be inconsistent among some replicas during the update process, but ensures that updates will eventually be propagated to all replicas. This makes it difficult to strictly maintain the ACID guarantees, as the "C" (consistency) part of ACID is sacrificed to provide reasonable availability [3].

In systems that host sensitive resources, accesses are protected via authorization policies that describe the conditions under which users should be permitted access to resources. These policies describe relationships between the system principles, as well as the certified credentials that users must provide to attest to their attributes.

The concept of *trusted transactions* is formalized. Trusted transactions are those transactions that do not violate credential or policy inconsistencies over the lifetime of the transaction.

Relaxed Consistency Models for the Cloud: Many database solutions have been written for use within the cloud environment. For instance, Amazon's Dynamo database; Google's BigTable storage system; Facebook's Cassandra; and Yahoo!'s PNUTS.

The common thread between each of these custom data models is the relaxed notion of consistency provided to support massively parallel environments.

Such a relaxed consistency model adds a new dimension to the complexity of the design of large scale applications and introduces a new set of consistency problems [5]. The authors of [6] presented a model that allows queriers to express consistency and concurrency constraints on their queries that can be enforced by the DBMS at runtime.

On the other hand, [7] introduces a dynamic consistency rationing mechanism that automatically adapts the level of consistency at runtime. Both of these works focus on data consistency, while our work focuses on attaining both data and policy consistency.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3, Special Issue 1, April 2015

Reliable Outsourcing: Security is considered one of the major obstacles to a wider adoption of cloud computing. Particular attention has been given to client security as it relates to the proper handling of outsourced data.

For example, proofs of data possession have been proposed as a means for clients to ensure that service providers actually maintain copies of the data that they are contracted to host [8]. In other words, data replication has been combined with proofs of retrievability to provide users with integrity and consistency guarantees when using cloud storage [9], [10].

To protect user access patterns from a cloud data store, Williams et al. [11] introduce a mechanism by which cloud storage users can issue encrypted reads, writes, and inserts. Further, Williams et al. [12] propose a mechanism that enables untrusted service providers to support transaction serialization, backup, and recovery with full data confidentiality and correctness. This work is orthogonal to the problem that we focus on in this paper, namely consistency problems in policy-based database transactions.

Distributed Transactions: Cloud TPS provides full ACID properties with a scalable transaction manager designed for a NoSQL environment [13]. However, Cloud TPS is primarily concerned with providing consistency and isolation upon data without regard to considerations of authorization policies. There has also been recent work that focuses on providing some level of guarantee to the relationship between data and policies [14]. This work proactively ensures that data stored at a particular site conforms to the policy stored at that site. If the policy is updated, the server will scan the data items and throw out any that would be denied based on the revised policy. It is obvious that this will lead to an eventually consistent state where data and policy conform, but this work only concerns itself with local consistency of a single node, not with transactions that span multiple nodes.

Distributed Authorization: The consistency of distributed proofs of authorization has previously been studied, though not in a dynamic cloud environment (e.g., [4]). This work highlights the inconsistency issues that can arise in the case where authorization policies are static, but the credentials used to satisfy these policies may be revoked or altered. The authors develop protocols that enable various consistency guarantees to be enforced during the proof construction process to minimize these types of security issues. These consistency guarantees are similar to the notions of safe transactions.

II. SYSTEM ANALYSIS

A. *Existing System:*

Interesting consistency problems can arise as transactional database systems are deployed in cloud environments and use policy-based authorization systems to protect sensitive resources. In addition to handling consistency issues among database replicas, it must also handle two types of security inconsistency conditions. First, the system may suffer from policy inconsistencies during policy updates due to the relaxed consistency model underlying most cloud services. For example, it is possible for several versions of the policy to be observed at multiple sites within a single transaction, leading to inconsistent (and likely unsafe) access decisions during the transaction. Second, it is possible for external factors to cause user credential inconsistencies over the lifetime of a transaction [4]. For instance, a user's login credentials could be invalidated or revoked after collection by the authorization server, but before the completion of the transaction.

B. *Proposed System:*

It begins by defining the notion of trusted transactions when dealing with proofs of authorization. Trusted transactions are those transactions that do not violate credential or policy inconsistencies over the lifetime of the transaction. Several different levels of policy consistency constraints and corresponding enforcement approaches are defined to guarantee the trustworthiness of transactions executing on cloud servers. A Two-Phase Validation Commit protocol is proposed as a solution, which is a modified version of the basic Two-Phase Validation Commit protocols.

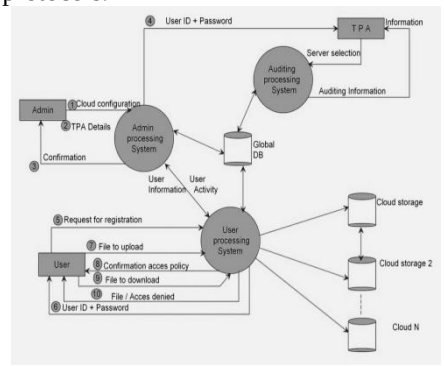

Fig 1: Architecture Diagram

As the name implies, 2PV operates in two phases: collection and validation. During collection, the TM first sends a Prepare-to-Validate message to each participant server. In response to this message, each participant 1) the YES or NO reply for the satisfaction of integrity constraints as in 2PC, 2) the TRUE or FALSE reply for the satisfaction of the proofs of authorizations, and 3) the version number of the policies used to build the proofs (v_i; p_i) as in 2PV.

Further, each participant keeps track of its reply (i.e.,the state of each query) which includes the id of the TM (TM_{id}), the id of the transaction (T_{id}) to which the query belongs, and a set of policy versions used in the query's authorization (v_i; p_i). Once the TM receives the replies from all the partici-pants, it moves on to the validation phase. If all polices are consistent, then the protocol

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3, Special Issue 1, April 2015

honors the truth value where any FALSE causes an ABORT decision and all TRUE cause a CONTINUE decision. In the case of inconsistent policies, the TM identifies the latest policy and sends an Update message to each out-of-date participant with a policy identifier and returns to the collection phase. In this case, the participants 1) update their policies, 2) re evaluate the proofs and, 3) send a new reply to the TM.

III. MODULE DESCRIPTION

Front End: Java
Client Interface Design: JSP and Servlets

Module 1:
Administrator: This module is concerned with the configuration and management of cloud storage and third party auditing system. The functions of administrator are listed below:

1) Cloud Storage Management (Add, View & Edit)
2) IP Configuration(Add, View & Edit)
3) TPA Management System

Module 2:
User: Each user must be registered and authorized to upload and download the files. The functions performed by an user are as follows:

1) Registration
2) Login (based on IP address user redirected to the server)
3) File Upload

- Browse and select the file to be uploaded
- Transfer the file to local server
- Create Hash Tag and send Hash tag and file details to Global Server
- Upload the file to Server Cloud storage
- Input Access Policy

1) View Uploaded File Details
2) File Download

- Select the file to be downloaded
- Check the Access Policy if Fail Stop the process
- Check the file availability in local server (if available)
- Change Password

1) Download history
2) Logout

Module 3:
Third Party Auditor (TPA): The TPA module is used to check the consistency of the cloud transactions by performing following functions in sequence:

1) Login
2) View Uploaded File Details
3) Third Party Auditing:

- Get all the files hash code in the selected server
- Compare the received hash code with global hash code table.
- If comparison fails for any one file then find the server which has the latest instance of the file.
- Copy the Latest Instance to the Auditing server.

1) Logout

IV. CONCLUSION

Despite the popularity of cloud services and their wide adoption by enterprises and governments, cloud providers still lack services that guarantee both data and access control policy consistency across multiple data centers. This paper identifies several consistency problems that can arise during cloud-hosted transaction processing using weak consistency models, particularly if policy-based authorization systems are used to enforce access controls. The policy based authorization technique ensures that the data can be only be accessed by users who have proper access privileges. System architecture has been designed with minimal cloud configuration and management of users and TPA by the administrator. Implementation and results are yet to be carried out in further stages.

REFERENCES

[1]. M. Armbrust et al., "Above the Clouds: A Berkeley View of Cloud Computing," technical report, Univ. of California, Feb. 2009.
[2]. S. Das, D. Agrawal, and A.E. Abbadi, "Elastras: An Elastic Transactional Data Store in the Cloud," Proc. Conf. Hot Topics in Cloud Computing (USENIX HotCloud '09), 2009.
[3]. D.J. Abadi, "Data Management in the Cloud: Limitations and Opportunities," IEEE Data Eng. Bull., vol. 32, no. 1, pp. 3-12, Mar. 2009.
[4]. A.J. Lee and M. Winslett, "Safety and Consistency in Policy-Based Authorization Systems," Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06), 2006.
[5]. W. Vogels, "Eventually Consistent," Comm. ACM, vol. 52, pp. 40-44, Jan. 2009.
[6]. H. Guo, P.-A. Larson, R. Ramakrishnan, and J. Goldstein, "Relaxed Currency and Consistency: How to Say "Good Enough" in SQL," Proc. ACM Int'l Conf. Management of Data (SIGMOD '04), 2004.
[7]. T. Kraska, M. Hentschel, G. Alonso, and D. Kossmann, "Consistency Rationing in the Cloud: Pay Only When It Matters," Proc. VLDB Endowment, vol. 2, pp. 253-264, Aug. 2009.
[8]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), 2007.
[9]. K.D. Bowers, A. Juels, and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), 2009.
[10]. A. Shraer, C. Cachin, A. Cidon, I. Keidar, Y. Michalevsky, and D. Shaket, "Venus: Verification for Untrusted Cloud Storage," Proc. ACM Workshop Cloud Computing Security (CCSW '10), 2010.
[11]. P. Williams, R. Sion, and B. Carbunar, "Building Castles Out of Mud: Practical Access Pattern Privacy and Correctness on Untrusted Storage," Proc. 15th ACM Conf. Computer and Comm. Security (CCS '08), 2008.
[12]. P. Williams, R. Sion, and D. Shasha, "The Blind Stone Tablet: Outsourcing Durability to Untrusted Parties," Proc. 16th Annual Network and Distributed System Security Symp. (NDSS '09), 2009.
[13]. Z. Wei, G. Pierre, and C.-H. Chi, "Scalable Transactions for Web Applications in the Cloud," Proc. 15th Int'l Euro-Par Conf. Parallel Processing (Euro-Par '09), Aug. 2009.
[14]. T. Wobber, T.L. Rodeheffer, and D.B. Terry, "Policy-Based Access Control for Weakly Consistent Replication," Proc. ACM Fifth European Conf. Computer Systems (EuroSys '10), 2010.