ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Two Phase Validation Commit Protocol to Ensure Balancing Performance, Accuracy, and Precision for Secure Cloud Transactions.

Hajera Tahseen,
M.Tech Student,
Software Engineering,
Lords Institute of Engineering And Technology.

Abdul Majeed,
Associate Professor,
HoD,
Lords Institute of Engineering And Technology.

ABSTRACT

Analyze in cloud computing be within the receipt of distributed process in transacting information utilize over cloud servers entities add the proof of authorization that area unit given Associate in Nursing explanation for assortment certified proof of authority. The proof and status it's corrected and collects over the exhausted some extent of time length below the threat of method Associate in Nursing authority policy of the shopper assured existence not obtainable circumstances. In this paper we tend to concentrate on the felt finding of the matter we tend to area unit defining the traditional understanding trusty group action after we are handling the proof of authorization in cloud computing and users will acquire there computation and storage to servers and it\'s additionally referred to as cloud. Cloud will create obtainable to be used different like application ex (Google apps, what's app) an oversized amount of knowledge keep in clouds it's extremely fast to notice security and privacy. it\'s additional vital issues in cloud computing. The users should evidence itself before initialization of any group action it should be certify that the cloud area unit different shoppers the cloud computing it keep the shoppers accounts is that the facts it outsources the cloud computing itself responsible to the service half from the specialist answer create sure the secure and lack of disturbance and it's additionally need for code implementation.

Index Terms—Cloud databases, authorization policies, consistency, distributed transactions, atomic commit protocol

INTRODUCTION

Cloud computing is associate expression uses of report a unique top computing concepts that needs a highest members of computers attach between actual time communication networks within the web. Cloud computing is recent technology comparison to the distributed computing within the networks. And methodology they capability to implement a programmed or application on take issue ent hooked up computers at the particular time they expression additionally a lot of oftentimes mention to web work base services that became visible to be assuming by real time atmosphere and therefore the reality pay up by virtual exertions reproduce by software package implementing on one or totally different real time machines equivalent virtual servers do not substantial be existing and it is forwarded around and scaled between up and down on the operator within the absence of influence they end-users to moderate extent sort of a cloud technology within the normal usage they word cloud crucial a image of the web dealers have in addition popularized they cloud computing technology it refers to the software platform and below structure that area unit sold as a service remotely through the web. Typically, the vendor has actual energy consuming servers that host merchandise and services from an overseas location, therefore end-users haven't got they will simply go surfing to the network while not putting in something. The major models of cloud computing service area unit referred to as Software as a Service, Platform as a Service, and Infrastructure as a Service. These cloud services is also offered in an exceedingly Public, personal or Hybrid

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

network. Google, Inc. is one of the foremost well-known cloud vendors. Cloud computing relies on sharing of resources to attain coherence and economies of scale, kind of like a utility (like the electricity grid) over a network. At the inspiration of cloud computing is the broader idea of converged infrastructure and shared services . The cloud additionally focuses on increasing the effectiveness of the shared resources. Cloud resources area unit usually not solely shared by multiple users however are dynamically reallocated per demand. this will work for allocating resources to users. for instance, a cloud pc facility that serves European users throughout European usiness hours with a selected application (e.g., email) could re assign the same resources to serve North yank users throughout North America's business hours with a unique application (e.g., an internet server). This approach ought to maximize the utilization of computing powers therefore reducing environmental harm as well since less power, aircon, rack space, etc. is required for a spread of functions. Proponents claim that cloud computing permits corporations to avoid direct infrastructure prices, and concentrate on comes that differentiate their businesses rather than infrastructure. Proponents additionally claim that cloud computing permits enterprises to induce their applications up and running quicker, with improved manageability and fewer maintenance, and allows IT to a lot of rapidly alter resources to fulfill unsteady and unpredictable business demand.

LITERATURE REVIEW

Enforcing Policy and Data Consistency of Cloud Transactions:-

In distributed transactional database systems deployed over cloud servers, entities cooperate to form proofs of authorizations that are justified by collections of certified credentials. These proofs and credentials may be evaluated and collected over extended time periods under the risk of having the underlying authorization policies or the user credentials being in inconsistent states. It therefore becomes possible for a policy - based authorization systems to make unsafe decisions that might threaten sensitive resources. This paper highlights the criticality of the problem. It then presents the first formalization of the concept of trusted transactions when dealing with proofs of authorizations. Accordingly, it defines different levels of policy consistency constraints and present different enforcement approaches to guarantee the trustworthiness of transactions executing on cloud servers. It proposed a Two-Phase Validation Commit protocol as a solution, that is a modified version of the basic Two-Phase Commit protocols. It finally provides performance analysis of the different presented approaches to guide the decision makers in which approach to use.

Data Management in the Cloud: Limitations and Opportunities;-

Recently the cloud computing paradigm has been receiving significant excitement and attention in the media and blogosphere. To some, cloud computing seems to be little more than a marketing umbrella, encompassing topics such as distributed computing, grid computing, utility computing, and software-as-a-service, that have already received significant research focus and commercial implementation. Nonetheless, there exist an increasing number of large companies that are offering cloud computing infrastructure products and services that do not entirely resemble the visions of these individual component topics. This article discussed the limitations and opportunities of deploying data management issues on these emerging cloud computing platforms (e.g., Amazon Web Services). It speculate that large scale data analysis tasks, decision support systems, and application specific data marts are more likely to take advantage of cloud computing platforms than operational, transactional database systems (at least initially). It present a list of features that a DBMS designed for large scale data analysis tasks running on an Amazon-style offering should contain. It discuss some currently available open source and commercial database options that can be used to perform such analysis tasks, and conclude that none of these options, as presently architected, match the requisite features. It

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

expressed the need for a new DBMS, designed specifically for cloud computing environments.

Automated Trust Negotiation Using Cryptographic Credentials:-

In automated trust negotiation (ATN), two parties exchange digitally signed credentials that contain attribute information to establish trust and make access control decisions. Because the information in question is often sensitive, credentials are protected according to access control policies. In traditional ATN, credentials are transmitted either in their entirety or not at all. This approach can at times fail unnecessarily, either because a cyclic dependency makes neither negotiator willing to reveal her credential before her opponent, because the opponent must be authorized for all attributes packaged together in a credential to receive any of them, or because it is necessary to fully disclose the attributes, rather than merely proving they satisfy some predicate (such as being over 21 years of age). Recently, several cryptographic credential schemes and associated protocols have been developed to address these and other problems. However, they can be used only as fragments of an ATN process. This paper introduces a framework for ATN in which the diverse credential schemes and protocols can be combined, integrated, and used as needed. A policy language is introduced that enables negotiators to specify authorization requirements that must be met by an opponent to receive various amounts of information about certified attributes and the credentials that contain it. The language also supports the use of uncertified attributes, allowing them to be required as part of policy satisfaction, and to place their (automatic) disclosure under policy control.

Distributed Proving in Access-Control Systems:-

This paper presents a distributed algorithm for assembling a proof that a request satisfies an access-control policy expressed in a formal logic, in the tradition of Lampson et al. It show analytically that our distributed proof-generation algorithm succeeds in assembling a proof whenever a centralized prover utilizing remote certificate retrieval would do so. In addition, we show empirically that our algorithm outperforms centralized approaches in various measures of performance and usability, notably the number of remote requests and the number of user interruptions. It shows that when combined with additional optimizations including caching and automatic tactic generation, which it introduce here, our algorithm retains its advantage, while achieving practical performance. Finally, it briefly describes the utilization of these algorithms as the basis for an access-control framework being deployed for use at this institution.

Policy-based Access Control for Weakly Consistent Replication:-

Enforcing authorization policy for operations that read and write distributed datasets can be tricky under the simplest of circumstances. Enforcement is too often dependent on implementation specifics and on policy detail that is inextricable from the data under management.

When datasets are distributed across replicas in a weakly-consistent fashion, for example when updates to policy and data propagate lazily, the problem becomes substantially harder. Specifically, if disjoint replicas can make different decisions about the permissibility of a potential modification due to temporary policy inconsistencies, then permanently divergent state can result. This paper describes and evaluates the design and implementation of an access-control system for weakly consistent replication where peer replicas are not uniformly trusted. This system allows for the specification of fine-grained access control policy over a collection of replicated items. Policies are expressed using a logical assertion framework and access control decisions are logical proofs. Policy can grow organically to encompass new replicas through delegation. Eventual consistency is preserved despite the fact that access control policy can be temporarily inconsistent.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Venus: Verification for Untrusted Cloud Storage:-

This paper presents Venus, a service for securing user interaction with un trusted cloud storage. Specifically, Venus guarantees integrity and consistency for applications accessing a key-based object store service, without requiring trusted components or changes to the storage provider. Venus completes all operations optimistically, guaranteeing data integrity. It then verifies operation consistency and notifies the application. Whenever either integrity or consistency is violated, Venus alerts the application. It implemented Venus and evaluated it with Amazon S3 commodity storage service. The evaluation shows that it adds no noticeable overhead to storage operations. It presents Venus, short for VErificatioN for Untrusted Storage. With Venus, a group of clients accessing a remote storage provider benefits from two guarantees: integrity and consistency. Integrity means that a data object read by any client has previously been written by some client; it protects against simple data modifications by the provider, whether inadvertent or caused by malicious attack. Note that a malicious provider might also try a \replay attack" and answer to a read operation with properly authenticated data from an older version of the object, which has been superseded by a newer version. Venus restricts such behavior and guarantees that either the returned data is from the latest write operation to the object, ensuring that clients see atomic operations, or that the provider misbehavior is exposed.

EXISTING SYSTEM

- To protect user access patterns from a cloud data store, Williams et al. introduce a mechanism by which cloud storage users can issue encrypted reads, writes, and inserts. Further, Williams et al. propose a mechanism that enables un trusted service providers to support transaction serialization, backup, and recovery with full data confidentiality and correctness.
- 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.
- 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. In other works, data replications have been combined with proofs of retrieve ability to provide users with integrity and consistency guarantees when using cloud storage.
- CloudTPS is primarily concerned with providing consistency and isolation upon data without regard to considerations of authorization policies.
- 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.
- The consistency of distributed proofs of authorization has previously been studied, though not in a dynamic cloud environment. 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.
- Disadvantages:-This Existing Works only focus on data consistency. It does not focus on policy consistency. This work only concerns itself with local consistency of a single node, not with transactions that span multiple nodes. 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.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

PROPOSED SYSTEM

- In this paper highlight the criticality of the problem. It defines the notion of trusted transactions when dealing with proofs of authorization. Accordingly, it propose several increasingly stringent levels of policy consistency constraints, and present different enforcement approaches to guarantee the trustworthiness of transactions executing on cloud servers.
- It proposed a Two-Phase Validation Commit protocol as a solution, which is a modified version of the basic Two-Phase Validation Commit protocols.
- It finally analyze the different approaches presented using both analytical evaluation of the overheads and simulations to guide the decision makers to which approach to use.
- In this paper address this confluence of data, policy, and credential inconsistency problems that can emerge as transactional database systems are deployed to the cloud.
- This paper formalized the concept of trusted transactions. Trusted transactions are those transactions that do not violate credential or policy inconsistencies over the lifetime of the transaction.
- It present a more general term, safe transactions, that identifies transactions that are both trusted and conforms to the ACID properties of distributed database systems.
- It defines several different levels of policy consistency constraints and corresponding enforcement approaches that guarantee the trustworthiness of transactions executing on cloud servers.
- It proposed a Two-Phase Validation Commit (2PVC) protocol that ensures that a transaction is safe by checking policy, credential, and data consistency during transaction execution.
- Advantages:- It provides a good balance between accuracy and performance, at the cost of higher code complexity.

IMPLEMENTATION

Cloud Formation:-

- First create a cloud infrastructure. It consisting of a set of S servers, where each server is responsible for hosting a subset of all data items belonging to a specific application domain.
- Users interact with the system by submitting queries or update requests encapsulated in ACID transactions. A transaction is submitted to a Transaction Manager (TM) that coordinates its execution.
- Multiple TMs could be invoked as the system workload increases for load balancing, but each transaction is handled by only one TM.
- It denote by the set of all credentials, which are issued by the Certificate Authorities (CAs) within the system. Here each CA offers an online method that allows any server to check the current status of credentials.

Two-Phase Commit (2PC) Algorithm: -

The 2-phase commit (2PC) protocol is a distributedalgorithm to ensure the consistent termination of a transactionin a distributed environment. Thus, via 2PC a unanimousdecision is reached and enforced among multiple participatingservers whether to commit or abort a given transaction, therebyguaranteeing atomicity. The protocol proceeds in two phases,namely the prepare (or voting) and the commit (or decision)phase, which explains the protocol's name.The protocol is executed by a coordinator process, whilethe participating servers are called participants. When thetransaction's initiator issues a request to commit thetransaction, the coordinator starts the first phase of the 2PCprotocol by querying—via prepare messages—all participantswhether to abort or to commit the transaction. If all participantsvote to commit then in the second phase the coordinatorinforms all participants to commit their share of the transactionby sending a commit message. Otherwise, the coordinatorinstructs all participants to abort their share of the transactionby sending an abort message.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Appropriate log entries arewritten by coordinator as well as participants to enable restartprocedures in case of failures.As long as a transaction is still executing ordinaryoperations, coordinators as well as all participants operate inthe Initial state. When the coordinator is requested to committhe transaction, it initiates the first phase of the 2PC protocol: To capture the state of the protocol's execution (which needs tobe available in case of protocol restarts as explained below),the coordinator first forces a begin log entry, which includes atransaction identifier as well as a list of the transaction's participants, to a stable log.

DATA OWNER REGISTERED WITH AUTHORIZATION POLICIES:-

- Next Data Owner Registered with authorization policies, valid date from and valid date to in desirable Trusted Third Party or CA.
- This Trusted Third Party or CA allows any server to check the current status of credentials.
- Then the CA creates secret keys for each data owner and end user. Because this Secret Keys are used to Authentication Purpose.
- A Data Owner wants to upload his file and end user wants to download a file, both are used this secret key for encryption and decryption.

UPLOAD FILE:-

- Data Owner wants to upload a file. So he encrypted this file using TA's secret Key.
- First he sends a key request to Trusted Third Party.
- Trusted Third Party creates a secret key and provide to Data Owner.
- Then the data owner encrypts his file using this secret key.

SAFE TRANSACTION:-

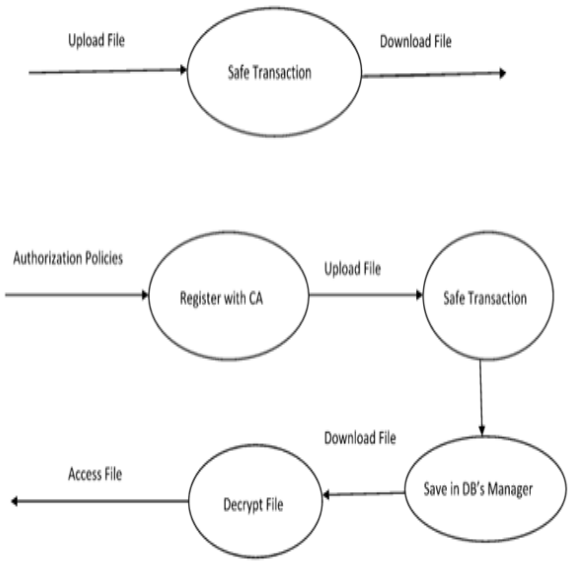
- A safe transaction is a transaction that is both trusted (i.e., satisfies the correctness properties of proofs of authorization) and database correct (i.e., satisfies the data integrity constraints).
- It first describes an algorithm that enforces trusted transactions (2PV), and then expands this algorithm to enforce safe transactions (2PVC).
- 2PV algorithm 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) evaluates the proofs for each query of the transaction using the latest policies it has available and 2) sends a reply back to the TM containing the truth value (TRUE/FALSE) of those proofs along with the version number and policy identifier for each policy used.
- Further, each participant keeps track of its reply (i.e., the state of each query) which includes the id of the TM, the id of the transaction to which the query belongs, and a set of policy versions used in the query's authorization.
- Once the TM receives the replies from all the participants, it moves on to the validation phase. If all polices are consistent, then the protocol 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) reevaluate the proofs and, 3) send a new reply to the TM.
- 2PVC can be used to ensure the data and policy consistency requirements of safe transactions.
- Specifically, 2PVC will evaluate the policies and authorizations within the first, voting phase. That is, when the TM sends out a Prepare-to-Commit message for a transaction,

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

the participant server has three values to report 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 as in 2PV, and 3) the version number of the policies used to build the proofs as in 2PV.

- The process for the TM under view consistency. It is similar to that of 2PV with the exception of handling the YES or NO reply for integrity constraint validation and having a decision of COMMIT rather than CONTINUE. The TM enforces the same behavior as 2PV in identifying policies inconsistencies and sending the Update messages. The same changes to 2PV can be made here to provide global consistency by consulting the master policies server for the latest policy version.

DATA FLOW DIAGRAM:



DOWNLOAD FILE:-

- An end User wants to access this upload file, he give the download request to particular DB's Server.
- This request contains filename, data owner and so on.
- The particular Server match this request to its database then retrieve the result and provide output to the user.
- Finally, the end users decrypt this file with data owner's secret key and access this file.

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. In this article, we identified 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. To this end, we developed a variety of light-weight proof enforcement and consistency models—i.e., Deferred, Punctual, Incremental, and Continuous proofs, with view or global consistency—that can enforce increasingly strong protections with minimal runtime overheads.

We used simulated workloads to experimentally evaluate implementations of our proposed consistency models relative to three core metrics: transaction processing performance, accuracy (i.e., global vs. view consistency and recency of policies used), and precision (level of agreement among transaction participants). We found that high performance comes at a cost: Deferred and Punctual proofs had minimal overheads, but failed to detect certain types of consistency problems. On the other hand, high accuracy models (i.e., Incremental and Continuous) required higher code complexity to implement correctly, and had only moderate performance when compared to the lower accuracy schemes. To better explore the differences between these approaches, we also carried out a trade-off analysis of our schemes to illustrate how application-centric requirements influence the applicability of the eight protocol variants explored in this article.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

REFERENCES

1. M.K. Iskander, D.W. Wilkinson, A.J. Lee, and P.K. Chrysanthis, "Enforcing Policy and Data Consistency of Cloud Transactions," Proc. IEEE Second Int'l Workshop Security and Privacy in Cloud Computing (ICDCS-SPCCICDCS-SPCC), 2011.

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. J. Li, N. Li, and W.H. Winsborough, "Automated Trust Negotiation Using Cryptographic Credentials," Proc. 12th ACM Conf. Computer and Comm. Security (CCS '05), Nov. 2005.

5. J. Li and N. Li, "OACerts: Oblivious Attribute Based Certificates," IEEE Trans. Dependable and Secure Computing, vol. 3, no. 4, pp. 340-352, Oct.-Dec. 2006.

6. L. Bauer et al., "Distributed Proving in Access-Control Systems," Proc. IEEE Symp. Security and Privacy, May 2005.

7. 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.

8. 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.