ABSTRACT

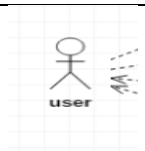
Cloud servers are used to utilize the distributed transactional database systems, entities are collaborated to form testament of authorization as that are justified by collections of certified credentials. These proofs and credentials may be evaluated and collected over extended time under the risk of having the underlying authorization protocol or the user credentials being in inconsistent states. Therefore, it is possible for Policy-Based Authorization Systems which results in unsafe decisions, that might threaten sensitive resources. The criticality of the problem is highlighted, we define the trusted transactions while dealing with proofs of authorization. Accordingly, we propose several increasingly valid levels of policy consistency constraints, and present different enforcement approaches to guarantee the trust worthiness of transactions executing on cloud servers. Two-Phase Validation Commit protocol is proposed as a solution. We finally analyze the different approaches presented using both analytical (systematic) evaluation (estimation) of the overheads and match to guide the decision maker to which approach to use.

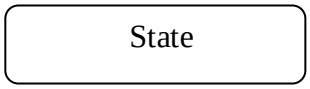
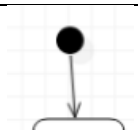
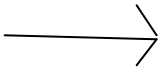
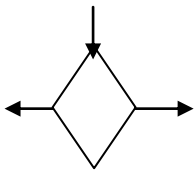
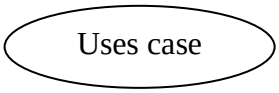
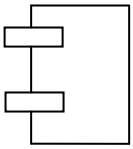
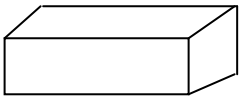
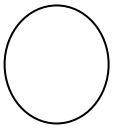
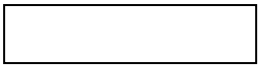
Keywords: Cloud Servers, Database systems, Authorization, Protocols, Inconsistent states, Accuracy.

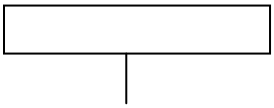
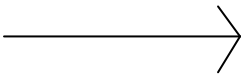
LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

S.NO	NOTATION NAME	NOTATION	DESCRIPTION
1	Class	+ public -private Class Name -attribute -attribute	Represents a collection of similar entities grouped together.
2	Association	Class A NAME Class B Class A Class B	Associations represents static relationships between classes. Roles represents the way the two classes see each other.
3	Actor		It aggregates several classes into a single classes.
4	Aggregation	Class A Class B Class A Class B	Interaction between the system and external environment
5	Relation (uses)	uses	Used for additional process communication.
6	Relation (extends)	extends →	Extends relationship is used when one use case is similar to another use case but does a bit more.
7	Communication	———	Communication between various use cases.

8	State		State of the process.
9	Initial State		Initial state of the object
10	Final state		Final state of the object
11	Control flow		Represents various control flow between the states.
12	Decision box		Represents decision making process from a constraint
13	Use case		Interact ion between the system and external environment.
14	Component		Represents physical modules which is a collection of components.
15	Node		Represents physical modules which are a collection of components.
16	Data Process/State		A circle in DFD represents a state or process which has been triggered due to some event or acion.
17	External entity		Represents external entities such as keyboard,sensors,etc.

18	Transition		Represents communication that occurs between processes.
19	Object Lifeline		Represents the vertical dimensions that the object communications
20	Message		Represents the message exchanged.

LIST OF ABBERIVATIONS

S.NO	ABBREVATION	EXPANSION
1.	DB	Data Base
2.	JVM	Java Virtual Machine
3.	JSP	Java Server Page
4.	CB	Collective Behavior
5.	SD	Social Dimension
6.	JRE	Java Runtime Environment
7.	SSD	Sparse Social Dimension
8.	LGP	Line Graph Partition

TABLE OF CONTENTS

Contents Page.no