File Exchange in a Private Cloud supported by a Trust Model

Edna Dias Canedo, Rafael Timóteo de Sousa Junior, Robson de Oliveira Albuquerque and Fábio Lúcio Lopes de Mendonça.
Electrical Engineering Department, University of Brasília – UNB – Campus Darcy Ribeiro – Asa Norte – Brasília – DF, Brazil, 70910-900.
E-mail: ednacanedo@unb.br, desousa@unb.br, robson@redes.unb.br, fabio.mendonca@latitude.eng.br

***Abstract*—Cloud computing is being progressively adopted in different business scenarios in order to obtain flexible and reliable computing environments, with several supporting solutions available in the market. Computing systems trust representation have been widely discussed and applied in a lot of information technology scenarios, becoming subject of scientific researches. This paper proposes the development of a trust model to ensure a reliable files exchange among nodes in a private cloud, as well as the calculation process of trust among them, according to the established metrics. The simulation result using CloudSim framework shows the effectiveness of the model in selecting more reliable node in private cloud.**

Keywords- cloudsim; cloud computing; private cloud; trust model; trust and security.

I. INTRODUCTION

Due to the development of new technologies in the broader fields of computing and the widespread use of the Internet, it has provided web applications that can be accessed independently of the location, by the internet. The development of virtualization technology enable sales on demand, in a scalable way, of computing resources and infrastructure, which are able to support web applications, arising this way cloud computing, generating an increasing tendency of applications that can be accessed efficiently, independent of the location. This way, it was born the necessity of rethink the way applications are developed and provided to users, while motivating the development of technologies to support their improvement.

Cloud computing is becoming more strengthful with the addition of large companies in the field of information technology, which has been making increasing efforts to develop technologies to this environment. Since the adoption of the cloud computing paradigm by IBM Corporation around the end of 2007, other companies of Information Technology is increasingly adopting cloud computing, for example, Google App Engine, which lets you create and host web applications with the same systems that actuate Google applications, Amazon Web Services of Amazon (AWS), Amazon Elastic Compute Cloud (EC2), Simple Storage Service (S3), Apple's iCloud and Microsoft's Azure Services Platform , that introduced products of cloud computing [1]. However, cloud computing still has risks related to data security in its different aspects, such as confidentiality, integrity, trust and authenticity [2] [3] [4].

This paper proposes a trust model to establish a ranking between nodes and enable trusted exchange of files between peers in a private cloud. Private cloud computing environment allows it to be working with a specific context of file distribution, so the files have a desired distribution and availability, being possible guarantees from the cloud manager that the access is restricted, and the identification of nodes is unique and controlled.

This paper is organized as it follows. In Section II, it's presented an overview of cloud computing, presenting a summary of its main features, architectures in cloud computing and deployment models. In Section III is presented the concepts of trust and review some related work about security, file system and trust in the cloud. In section IV is presented the file distribution in cloud. In section V, we introduce the proposed trust model and results and analysis of Simulation. Finally, in Section VI, we conclude with a summary of our results and directions for new research.

II. CLOUD COMPUTING

Cloud computing refers to the use, through the Internet, of diverse applications as if they were installed in the user's computer, independently of platform and location. Several formal definitions for cloud computing have been proposed by industry and academy. We adopt the following definition: "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [5].

Cloud computing is being progressively adopted in different business scenarios in order to obtain flexible and reliable computing environments, with several supporting solutions available in the market. Being based on diverse technologies (e.g. virtualization, utility computing, grid computing and service oriented architectures) and constituting a whole new computational paradigm, cloud computing requires high level management routines. Such management activities include: (a) service provider selection; (b) virtualization technology selection; (c) virtual resources allocation; (d) monitoring and auditing in order to guarantee Service Level Agreements (SLA).

A. Cloud Computing Architecture

Cloud computing architecture is based on layers. Each layer deals with a particular aspect of making application

978-0-7695-4810-4/12 $26.00 © 2012 IEEE
DOI 10.1109/CyberC.2012.23

resources available. Basically there are two main layers: a lower and a higher resource layer. The lower layer comprises the physical infrastructure and is responsible for the virtualization of storage and computational resources. The higher layer provides specific services, such as: software as service, platform as service and infrastructure as service. These layers may have their own management and monitoring system, independent of each other, thus improving flexibility, reuse and scalability. Figure 1 presents the cloud computing architectural layers [7] [8].

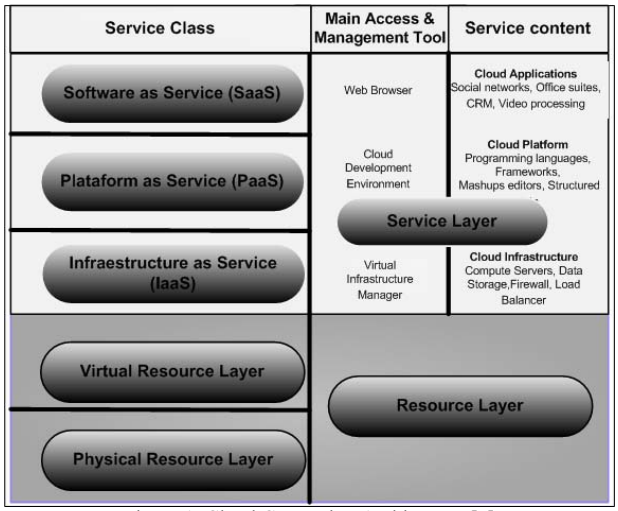

Figure 1. Cloud Computing Architecture [7]

B. Cloud Computing Deployment

According to the intended access methods and availability of cloud computing environments, there are different models of deployment [5]. Access restriction or permission depends on business processes, the type of information and characteristics of the organization. In some organizations, a more restrict environment may be necessary in order to ensure that only properly authorized users can access and use certain resources of the deployed cloud services. A few deployment models for cloud computing are discussed in this section. They include private cloud, public cloud, community cloud and hybrid cloud [5].

III. TRUST

The concepts of trust, trust models and trust management has been the object of several recent research projects. Trust is recognized as an important aspect for decision-making in distributed and auto-organized applications [14], [15]. In spite of that, there is no consensus in the literature on the definition of trust and what trust management encompasses. In the computer science literature, Marsh is among the first to study computational trust. Marsh [14] provided a clarification of trust concepts, presented an implementable formalism for trust, and applied a trust model to a distributed artificial intelligence (DAI) system in order to enable agents to make trust-based decisions.

Beth et al. [15] also proposed a trust model for distributed networks. They derived trust recommendations from direct trust and gave them formal representations, as well as rules to derive trust relationships and algorithms to compute trust values. Josang et al. [16] describe a trust model where positive and negative feedback about a specific member is accumulated. The model is based on the Bayesian network model, using the beta probability density function to calculate a member's expected future behavior.

Trust is considered to be more than the authorized nature of security relations between human societies, which achieve stable and healthy operation, to a large extent thanks to the trust relationship between the individuals, groups and organizations. Therefore, in a large number of dynamic user-oriented open network environments, the study of the trust relationships between the trust-based security mechanisms to ensure the safe operation of distributed applications has become a fundamental topic.

Currently, most scholars have reached a consensus that trust should have three important features [17]. 1) Subjectivity (different entities of the same view of things which will be affected by factors such as individual preferences may vary); 2) The expected probability (the degree of trust can be extracted and formalized as the estimated likelihood of a given event); 3) Relevance (trust is an aspect of things, for specific content).

In recent works on trust, mainly two distinct methods are used for subjective trust reasoning: probabilistic reasoning based on statistical hypothesis testing; and approaches based on fuzzy theory, expert systems and artificial intelligence techniques. However, these methods do not fully reflect the essential nature of trust. Subjective trust, in essence, is based on the belief that it has great uncertainty. In the subjective, objective world, random and fuzzy uncertainties are the two main forms that have become the industry consensus [18]. Thus, the axiomatic methods based on probability theory or fuzzy set theory doesn't achieve a comprehensive assessment of trust information.

A. Security in the Cloud

A number of technologies have been employed in order to provide security for cloud computing environments. The creation and protection of security certificates is usually not enough to ensure the necessary security levels in the cloud. Cryptographic algorithms used with cloud applications usually reduce performance and such reduction must be restricted to acceptable levels [10].

Cloud computing offers users a convenient way of sharing a large quantity of distributed resources belonging to different organizations. On the other hand, the very nature of the cloud computing paradigm makes security aspects quite more complex. Trust is the main concern of consumers and service providers in a cloud computing environment [11]. The inclusion of totally different local systems and users of quite diverse environments brings special challenges to the security of cloud computing. On one hand, security mechanisms must offer users a high enough level of guarantees. On the other hand, such mechanism must not be so complex, enough to make it difficult for users to use the system. The openness and computational flexibility of popular commercially available operating systems have been important factors to support the general adoption of cloud

computing. Nevertheless, these same factors increase system complexity, reduce the degree of trust and introduce holes that become threats to security [11].

Huan et al. [12] investigate the different security vulnerability assessment methods for cloud environments. Experiments show that more vulnerabilities are detected if vulnerable tools and servers are in the same LAN. In other word, the hackers can find an easier way to get the target information if it is on the same LAN of the compromised systems. Experimental results can be used to analyze the risk in third party compute clouds. Popovic et al. [4] discuss security issues, requirements and challenges that Cloud Service Providers (CSP) face during cloud engineering. Recommended security standards and management models to address these are suggested both for the technical and business community.

B. Filesystem

Uppoor et al. [3] present a new approach for synchronizing of hierarchically distributed file systems. Their approach resembles the advantages of peer-to-peer synchronization, storing online master replicas of the shared files. The proposed scheme provides data synchronization in a peer-to-peer network, eliminating the costs and bandwidth requirements usually present in cloud computing master-replica approaches.

The work in [13] presents CDRM, a scheme for dynamic distribution of file replicas in a cloud storage cluster. This scheme periodically updates the number and location of file block replicas in the cluster. The number of replicas is updated according to the actual availability of cluster nodes and the expected file availability. The dynamic distribution algorithm for replica placement takes into account the storage and computational capacity of the cluster nodes, as well as the bandwidth of the communication network.

C. Trust in the Cloud

Trust and security have become crucial to guarantee the healthy development of cloud platforms, providing solutions for concerns such as the lack of privacy and protection, the guarantee of security and author rights.

Privacy and security have been shown to be two important obstacles concerning the general adoption of the cloud computing paradigm. In order to solve these problems on the IaaS service layer, a model of trustworthy cloud computing which provides a closed execution environment for the confidential execution of virtual machines was proposed [19]. This work has shown how the problem can be solved using a Trusted Platform Module. The proposed model, called Trusted Cloud Computing Platform (TCCP), is supposed to provide higher levels of reliability, availability and security. In this solution, there is a cluster node that acts as a Trusted Coordinator (TC). Other nodes in the cluster must register with the TC in order to certify and authenticate its key and measurement list. The TC keeps a list of trusted nodes.

Zhidong et al. [11] presented a method for building a trustworthy cloud computing environment by integrating a Trusted Computing Platform (TCP) to the cloud computing system. The TCP is used to provide authentication, confidentiality and integrity [11]. This scheme displayed positive results for authentication, rule-based access and data protection in the cloud computing environment.

Cloud service providers (CSP) should guarantee the services they offer, without violating users' privacy and confidentiality rights. Li et al. [20] introduced a multi-tenancy trusted computing environment model (MTCEM). This model was designed for the IaaS layer with the goal of ensuring a trustworthy cloud computing environment to users. MTCEM has two hierarchical levels in the transitive trust model that supports separation of concerns between functionality and security. It has 3 identity flows: a) the consumers, who hire the CSP cloud computing services; b) the CSP, that provides the IaaS services; c) the auditor (optional, but recommended), who is responsible for verifying whether the infrastructure provided by the CSP is trustworthy on behalf of users. In MTCEM, the CSP and the users collaborate with each other to build and maintain a trustworthy cloud computing environment.

Zhimin et al. [13] propose a collaborative trust model for firewalls in cloud computing. The model has three advantages: a) it uses different security policies for different domains; b) it considers the transaction contexts, historic data of entities and their influence in the dynamic measurement of the trust value; and c) the trust model is compatible with the firewall and does not break its local control policies. A model of domain trust is employed. Trust is measured by a trust value that depends on the entity's context and historical behavior, and is not fixed. The cloud is divided in a number of autonomous domains and the trust relations among the nodes are divided in intra and inter-domain trust relations. The intra-domain trust relations are based on transactions operated inside the domain. Each node keeps two tables: a direct trust table and a recommendation list.

In [21] a trusted cloud computing platform (TCCP) which enables IaaS providers to offer a closed box execution environment that guarantees confidential execution of guest virtual machines (VMs) is proposed. This system allows a customer to verify whether its computation will run securely, before requesting the service to launch a VM. TCCP assumes that there is a trusted coordinator hosted in a trustworthy external entity. The TCCP guarantees the confidentiality and the integrity of a VM user, and allows a user to determine up front whether or not the IaaS enforces these properties.

The work [6] evaluates a number of trust models for distributed cloud systems and P2P networks. It also proposes a trustworthy cloud architecture (including trust delegation and reputation systems for cloud resource sites and datacenters) with guaranteed resources including datasets for on-demand services.

IV. FILE DISTRIBUTION IN CLOUD

Cloud computing offers great flexibility for users, due to the fact that users don't have to worry about management complexity related to each system, for example, the databases can be transferred to data centers of large

specialized companies, although the management data in outsourced environments aren't always reliable. Users are becoming dependent on the availability and integrity offered by storage service providers. Thus, it is necessary to use models of secure data storage in order to ensure the integrity of cloud user's data [22].

One of the problems that cloud computing is able to solve is the storage of files and their distribution with high rate of availability. There are several approaches to manage data in the cloud and each system uses a specific approach to persist data. Among these approaches, we can highlight new file systems, frameworks and proposals for storage and processing data.

A. Google File System

The Google File System (GFS) is a distributed file system proprietary developed by Google and specially designed to provide efficient and reliable access to data, using large server clusters [23]. The GFS architecture consists of three elements: Clients, Master and chunkservers.

A GFS cluster consists of a single master and multiple chunkservers that is accessed by multiple clients, as shown in Figure 2. The master stores three major types of metadata: file and chunk namespaces, mapping from files to chunks, and the locations of each chunk's replicas. All metadata is updated by the master server, which communicates regularly with each chunkserver through the exchange of messages called heartbeat messages, to give it instructions and collect its state.

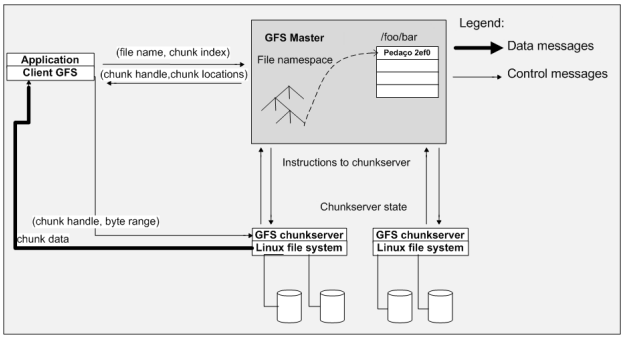

Figure 2. Architecture of GFS [23]

B. Amazon S3

The Amazon Simple Storage Service (S3) is a distributed storage system based on Dynamo [24]. Dynamo uses key-value model stored in a Distributed Hash Table (DHT) and has no support associations or schemes. To ensure a level of scalability and availability, data is partitioned and replicated in multiple machines, using a consistent hashing, being consistency facilitated by multiple versions of objects.

The consistency between replicas during updates is maintained by a quorum-like technique and a decentralized replica synchronization protocol. Dynamo employs a gossip based distributed failure detection and membership protocol. Dynamo is a completely decentralized system with minimal need for manual administration. Storage nodes can be added and removed from Dynamo without requiring any manual partitioning or redistribution.

Dynamo stores objects associated with a key through a simple interface; it exposes two operations: get() and put(). The get (key) operation locates the object replicas associated with the key in the storage system and returns a single object or a list of objects with conflicting versions along with a context. The put (key, context, object) operation determines where the replicas of the object should be placed based on the associated key, and writes the replicas to disk. The context encodes system metadata about the object that is opaque to the caller and includes information such as the version of the object.

C. Microsoft Azure

Microsoft SQL Azure is compound of a set of services for storing and processing data in cloud [25]. SQL Azure with Windows Azure Storage composes the solution of data management in cloud of Microsoft. The purpose of Windows Azure Storage is provide a scalable, durable, highly available storage and provides users the payment on demand. It allows easy access to data, providing a simple interface, remotely available and in data centers. The storage services in Windows Azure Storage are offered in four levels of abstraction: blobs, table, drives and queue structures. Windows Azure Storage includes persistent storage through blobs, tables and queues. The storage access and load balancing is done automatically through a set of nodes responsible for physical storage providing scalability and availability. To use Windows Azure Storage service, user needs to create a storage account, which can be obtained from the Windows Azure portal web interface. After the creation of an account, user will receive a 256-bit secret key.

D. Hadoop

Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware [26] and its objective is storing large amounts of data across multiple nodes. The Figure 3 shown architecture distributed file system HDFS.

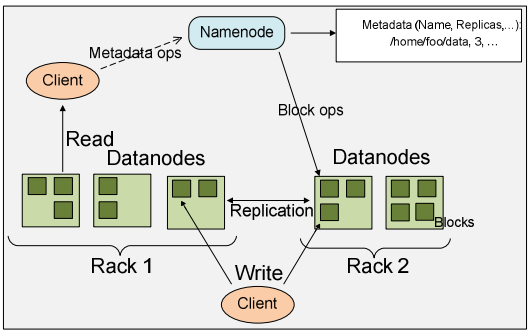

Figure 3. Architecture Distributed File System HDFS [26]

HDFS has master/slave architecture. An HDFS cluster consists of a single NameNode, a master server that manages the file system namespace and regulates access to files by clients. In addition, there are a number of DataNodes, usually one per node in the cluster, which manage storage attached

to the nodes that they run on. HDFS exposes a file system namespace and allows user data to be stored in files. Internally, a file is split into one or more blocks and these blocks are stored in a set of DataNodes. The NameNode executes file system namespace operations like opening, closing, and renaming files and directories. It also determines the mapping of blocks to DataNodes. The DataNodes are responsible for serving read and write requests from the file system's clients. The DataNodes also perform block creation, deletion, and replication upon instruction from the NameNode.

V. TRUST MODEL FOR PRIVATE CLOUD

According to the review and related research [2] [3] [7] [11] [13] [19] [27], it is necessary to employ a cloud computing trust model to ensure the exchange of files among cloud users in a trustworthy manner. In this section, we introduce a trust model to establish a ranking of trustworthy nodes and enable the secure sharing of files among peers in a private cloud. The environment computing private cloud was chosen because we work with a specific context of distributing files, where the files have a desired distribution and availability.

We propose a trust model where the selection and trust value evaluation that determines whether a node is trustworthy can be performed based on node storage space, operating system, link and processing capacity. For example, if a given client has access to a storage space in a private cloud, it still has no selection criterion to determine to which cloud node it will send a particular file. When a node wants to share files with other users, it will select trusted nodes to store this file through the proposed following metrics: processing capacity (the average workload processed by the node, for example, if the node's processing capacity is 100% utilized, it will take longer to attend any demands), operating system (operating system that has a history of lower vulnerability will be less susceptible to crashes), storage capacity and link (better communication links and storage resources imply greater trust values, since they increase the node's capacity of transmitting and receiving information). The trust value is established based on queries sent to nodes in the cloud, considering the metrics previously described.

Each node maintains two trust tables: direct trust table and the recommended list. a) If a node needs to calculate the trust value of another node, it first checks the direct trust table and uses the trust value if the value for the node exists. If this value is not available yet, then the recommended lists are checked to find a node that has a direct trust relationship with the desired node the direct trust value from this node's direct trust table is used. If there's no value attached, then it sends a query to its peers requesting information on their storage space, processing capacity and link. The trust values are calculated based on queries exchanged between nodes.

b) The requesting node will assign a greater trust value to nodes having greater storage capacity and / or processing and better link. In addition, the operating system will also be considered as a criterion of trust.

In this model is assumed that the node has a unique identity on the network. As trust is evolutionary, when a node joins the network, the requesting node doesn't know, soon it will be asked about his reputation to other network nodes. If no node has information about respective node (it has not had any experience with it), the requesting node will decide whether the requested relate to, initially asking some activity / demand for it to run. From its answers will be built trust with its node. Trust table node will contain a timer (saving behavior / events that raise and lower the trust of a given node) and will be updated at certain times.

Figure 4 presents a high level view the proposed trust model, where the nodes query their peers to obtain the information needed to build their local trust table.

In this model, a trust rank is established, allowing a node A to determine whether it is possible to trust a node B to perform storage operations in a private cloud. In order to determine the trust value of B, node A first has to obtain basic information about this node.

The Figure 5 presents scenario of information request for a reliable exchange of files between nodes. When node A needs to exchange a file in cloud and it wants to know if node B is trusted to send and store the file, it will use the proposed Protocol Trust Model, which can be described with the following scenario: step 1, node A sends a request to the nodes of cloud, including node B, asking about storage capacity, operating system, processing capacity and link.

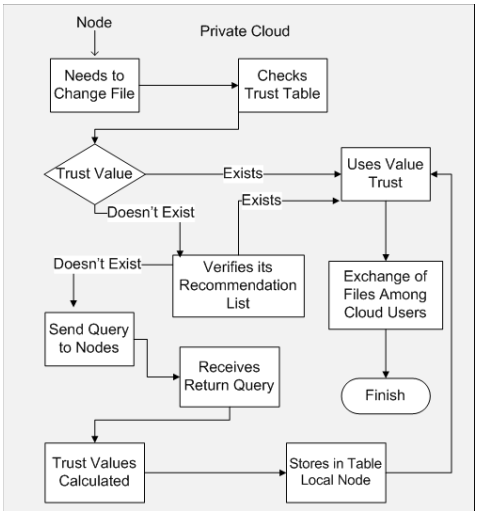

Figure 4. High Level Trust Model

In step 2, nodes, including node B, send a response providing the requested information. In step 3, node A evaluates the information received from B and from all nodes. If the information provided by B, are consistent with the expected, with the average value of the information of other nodes, the values are stored in local recommendations table of node A, after to make the calculation of trust and store in your local trust table.

The trust value of a node indicates its disposition/suitability to perform the operations between peers of cloud. This value is calculated based on the history

interactions/queries between the nodes, value ranging between [0, 1].

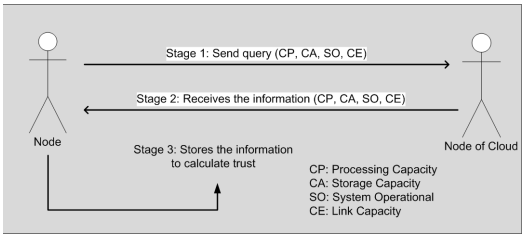

Figure 5. Scenario of Information Requests

In general, trust of node A in node B, in the context of a private cloud NP, can be represented by a value V which measures the expectation that a particular node will have good behavior in the private cloud, so trust can be expressed by:

$$T_{(a,b)}^{np} = V_{np}^{b} \quad (1)$$

$T_{(a,b)}^{np}$ represent the trust of A in B in the private cloud NP and V_{np}^{b} represent the trust value of B, in the private cloud NP analyzed by A. According to definition of trust, V_{np}^{b} is equivalent to queries sent and received (interaction) by A related to B in cloud NP. As the interactions are made between the nodes of private cloud, the information is used for the calculation of trust.

Nodes of a private cloud should be able to consider whether a trust value is acceptable, generating trust level. If the node exceeds the level within a set of analyzed values, it must be able to judge the node in a certain degree of trust. Trust degree can vary according to a quantitative evaluation: a node has a very high trust in another one, a node has low trust in another one, a node doesn't have sufficient criteria to opine, a node trusts enough to opine, etc. In our model, one node trusts another node from trust value $T \geq 0.6$ [14].

The trust values are calculated from queries between the nodes of NP, allowing obtaining the necessary information for final calculation of trust. The trust information is stored through the individual records of interaction with the respective node, staying in local database information about the behavior of each node in the cloud that wants to exchange a file (local trust table and local recommendations table).

A. Calculation of Trust

Four aspects can to have impact on calculation of direct trust of a node. Greater storage capacity and processing capacity have more weight in the choice of a node more reliable, because of these features are the responsible for ensure the integrity and file storage. To calculate direct trust of a node, it is attributed by administrator of the private cloud: storage capacity and processing with weights of 35%, 15% to link and the remaining 15% to operating system. Knowing that a node can to have the trust value ranging from [0.1] and that these values are variable over time, a node can have its storage capacity increased or decreased, it's necessary that trust reflects the behavior of a node in a given period of time. Nodes with constant characteristics should therefore be more reliable because they have less variation in basic characteristics.

According to the weights attributed it's possible to calculate the trust of node. The calculation of trust node A in B in cloud NP will be represented by:

$$T_{(a,b)}^{fnp} = \sum_{np=1}^{j} V_{np}^{b} \frac{((b,m_1)*0{,}35)+((b,m_2)*0{,}35)+((b,m_3)*0{,}15)+((b,m_4)*0{,}15) \leq 1}{j > 0} \quad (2)$$

$T_{(a,b)}^{fnp}$ represents the final trust of A in B in cloud NP. The trust value of B is defined as the sum of metrics values that the node B has (m) in the cloud NP; j represents the number of interactions of trust from node A in B in the cloud NP, where $j \geq 0$.

B. Description of the Environment Simulations

The simulations involving the model were performed using the framework CloudSim [28]. The simulated scenario includes an IaaS provider, which has three data centers and a client that offers this service. The client uses the resources offered by the provider for allocation and sending of virtual machine that executes a set of tasks, called cloudlets. The dynamic of choice of datacenters for allocation and sending of virtual machines and execution of cloudlets is defined by the profile of customer usage and the features offered by the provider. So the scenario simulated in this work consists of an IaaS provider that has three datacenters distributed in different locations, Goiânia-GO, Anapolis – GO and Brasília-DF, a customer with a usage profile, 04 hosts, 30 virtual machines and 100 cloudlets.

The processing architecture, the operating system, the hyper display used and the costs of use are unique attributes of a datacenter, and it is automatically applied to all its nodes. Moreover, the processing capacity, the capacity of system memory and storage capacity are characteristic of hosts that can be individually customized by creating datacenters with heterogeneous resources. The three simulated datacenters are composed of homogeneous nodes, so, all hosts on the same datacenter have the same configuration/feature.

The composition of the simulated scenario is composed of a client who initially creates 100 tasks (cloudlets). The client sends the cloudlets to the broker that manages for whom the tasks will be sent. The broker has at his disposal three datacenters, Anapolis, Goiania and Brasilia, which have virtual machines that perform the cloudlets and send it back after the execution. Virtual machines should be allocated to hosts that attend the minimum resources necessary. Hosts chosen must belong to a data center that attends the customer's usage profile.

C. Results and Simulations

If the CloudSim simulation environment is defined and configured, and since the weights are assigned to the metrics, it can be made the calculation of the trust of a node by running the scenarios implemented in the CloudSim framework [28]. The results are divided into three topics.

The first deals with the presentation of the results that all tasks / cloudlets were successfully executed. The second topic deals with the results and analysis that tasks / cloudlets were performed with success and failure. The third deals with the results and analysis of the simulated environment with changes in one of the metrics.

D. Simulation Scenario with all Tasks Successfully Performed

To perform the simulation of the proposed environment, it's initially necessary to define the settings that are considered ideal for a reliable machine, so it's necessary to define the baseline configuration of the machine in order to compare with the values of other virtual machines in the simulation environment. As in the context of this application there are small and low complexity tasks, it's used as baseline the standard configuration set by Amazon [29], trying to get closer as possible to the existing cost-benefit in real clouds, where the machine settings are compatible with the charges and offered services. The configuration used in this paper is presented in Table I.

TABLE I. MACHINE CONFIGURATION BASELINE [29]

Values	
deal Size for HD	163840 MB
Ideal Size for RAM	1740 MB
Ideal Size for MIPS	5000
Ideal Size for Bandwidth	1024 Kbytes

In order to make comparisons and analysis of the results in different scenarios, different simulations were performed during the proposed work.

Figure 6 shows the trust of each virtual machine after the running of the 100 cloudlets. Some machines didn't perform cloudlets because they don't satisfy the conditions of evaluation / verification of a reliable machine to perform a task, compared with the baseline machine. The virtual machines that successfully executed cloudlets were: VM 05, running 13 tasks, MV 6, running 13 tasks; VM 7 running 11 tasks; VM 15, running 12 tasks, VM 16, running 13 tasks, running 11 tasks, VM 17, running 12 tasks, VM 25, VM 26 running 13 tasks and VM 27 running 02 tasks. The other virtual machines didn't perform any tasks / cloudlet because they didn't have the desired trust level.

Figure 7 shows the trust level of Virtual Machine 16 that performed the highest number of cloudlets during the simulation, 13 tasks. The initial reliability value of the machine was 0.671875 and the final value 1.0.

The trust of a virtual machine in the simulated model increases in proportion as human being, example, when an individual performs an activity or solve a particular problem for us successfully, our trust is increased gradually. Thus, each cloudlet successfully executed, the trust value of a VM will be increased by 2.5%, until the trust level arrives at 0.85. Above 0.85, trust increases 5% until it reaches the maximum trust of 1.0.

If a machine doesn't perform a certain task successfully, it doesn't solve its problem, it loses trust. The weight of suspicion is usually greater than the weight of trust. Thus, in our simulated model the rate of suspicion is 5% for each task performed without success.

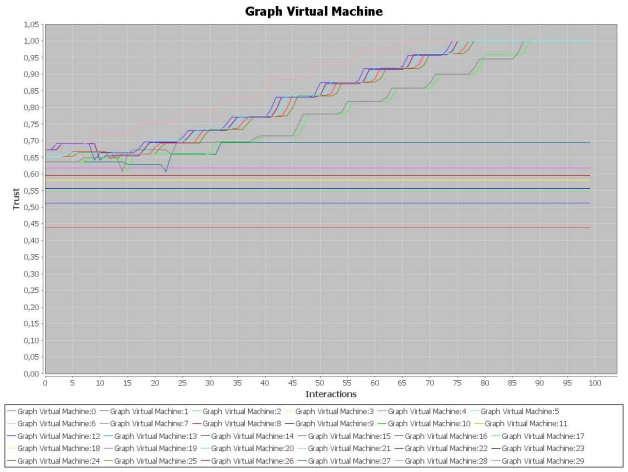

Figure 6. Trust of Virtual Machines After Execution

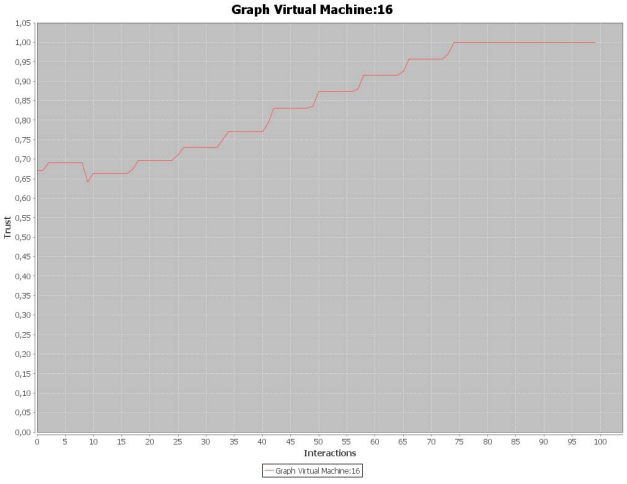

Figure 7. Trust of Virtual Machine 16 running 13 Cloudlets

Analyzing the results of the simulations which all tasks were performed successfully in the CloudSim environment, it's possible to identify the trust level of the virtual machines that performed the cloudlets. According to the reference information, a node trusts another in function of its trust value. As trust reflects the operation of trust between the nodes of a private cloud and the client and all virtual machines are above the reference value, they are considered reliable. Thus the trust of the node tends to increase over time as the successful interactions occur, as can be seen in the results.

VI. CONCLUSIONS

Cloud computing has been focus of several recent research, which demonstrates the importance and need of trust model that ensures reliable and secure exchange files.

Promising area to be explored through research and experimental analyzes, using the trust to mitigate the

computational problems in aspects related to security, trust and reputation to guarantee the exchange of information on private cloud environments, reducing the possibility of failure and or change information on file-sharing, involving metrics that are capable of representing or mapping the trust level of a network node to realize the exchange of files in a private cloud.

The simulations and results allow to identify which taken metrics directly influence the calculation of trust in a node. The future simulations using a real environment for private cloud computing will allow to evaluate the behavior of nodes as well as the history of their iterations and assumed values where there are tasks carried out with success and failure. The use of open platform, CloudSim [28], to perform simulations of the behavior of node, allowed to calculate the table of trust and select nodes considered more reliable. Furthermore, we evaluated the adequacy of the metrics used in the proposed trust model, allowing to identify and select the most appropriate in relation to the historical behavior of the nodes belonging to the analyzed environment.

REFERENCES

[1] Zhang Jian-jun and Xue Jing."A Brief Survey on the Security odel of Cloud Computing," 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES), Hong Kong IEEE, pp. 475 – 478, 2010.

[2] Wang Han-zhang and Huang Liu-sheng."An improved trusted cloud computing platform model based on DAA and Privacy CA scheme," IEEE International Conference on Computer Application and System Modeling (ICCASM 2010). 978-1-4244-7235-2, 2010.

[3] Uppoor, S., M. Flouris, and A. Bilas. "Cloud-based synchronization of distributed file system hierarchies," Cluster Computing Workshops and Posters (CLUSTER WORKSHOPS), IEEE International Conference, pp. 1-4. 2010.

[4] Popovic, K. and Z. Hocenski. "Cloud computing security issues and challenges," MIPRO, 2010 Proceedings of the 33rd International Convention, pp. 344-349, 24-28 May 2010.

[5] P. Mell and T. Grance, "Draft nist working definition of cloud computing - v15," 21. Aug 2009.

[6] Chang. E, T. Dillon and Chen Wu, "Cloud Computing: Issues and Challenges," 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), pp. 27-33. Australia, 2010.

[7] Xue Jing and Zhang Jian-jun, "A Brief Survey on the Security Model of Cloud Computing," 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES), Hong Kong IEEE, pp. 475 – 478. Aug 2010.

[8] Minqi Zhou, Rong Zhang, Dadan Zeng, and Weining Qian, "Services in the cloud computing era: a survey," Software Engineering Institute. Universal Communication. Symposium (IUCS), 4th International. IEEE Shanghai, pp. 40-46. China. 978-1-4244-7821-7 (2010).

[9] A. Marinos and G. Briscoe, "Community cloud computing," in First International Conference Cloud Computing, CloudCom, volume 5931 of Lecture Notes in Computer Science, pp. 472–484. Springer (2009).

[10] H. Takabi, J. B. D. Joshi, and G. Ahn, "Security and Privacy Challenges in Cloud Computing Environments," IEEE Security and Privacy, vol. 8, no. 6, pp. 24-31, Nov./Dec. 2010, doi:10.1109/MSP.2010.186.

[11] Zhidong Shen, Li Li, Fei Yan, and Xiaoping Wu, "Cloud Computing System Based on Trusted Computing Platform," Intelligent Computation Technology and Automation (ICICTA), IEEE International Conference on Volume: 1, pp. 942-945. China. 2010.

[12] Huan-Chung Li, Po-Huei Liang, Jiann-Min Yang, and Shiang-Jiun Chen, "Analysis on Cloud-Based Security Vulnerability Assessment," 2010 IEEE 7th International Conference on e-Business Engineering (ICEBE), pp. 490-494, 2010.

[13] Zhimin Y., Lixiang Q., Chang L.,Chi Y., and Guangming W,"A collaborative trust model of firewall-through based on Cloud Computing," Proceedings of the 2010 14th International Conference on Computer Supported Cooperative Work in Design. Shanghai, China. pp. 329-334, 14-16. 2010.

[14] S. P. Marsh, "Formalising Trust as a Computational Concept", Ph.D. Thesis, University of Stirling, 1994.

[15] T. Beth, M. Borcherding, and B. Klein, "Valuation of trust in open networks," In ESORICS 94. Brighton, UK, November 1994.

[16] A. Jφsang and S. J. Knapskog, "A metric for trusted systems," Global IT Security, pp. 541-549, 1998.

[17] A. Abdul-Rahman and S. Hailes, "A distributed trust model," In Proceedings of the 1997 New Security Paradigms Workshop, pp. 48-60, 1998.

[18] A. Jøsang and R. Ismail, "The Beta Reputation System," In Proceedings of the 15th Bled Electronic Commerce Conference, pp. 17-19. June 2002.

[19] Xiao-Yong Li, Li-Tao Zhou, Yong Shi, and Yu Guo, "A Trusted Computing Environment Model in Cloud Architecture," Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, 978-1-4244-6526-2. Qingdao, pp. 11-14. China. July 2010.

[20] Li Xiaoqi, Lyu M R, and Liu Jiangchuan. "A trust model based routing protocol for secure AD Hoc network," Proceedings of the 2004 IEEE Aerospace Conference, pp. 1286-1295. 2004.

[21] N. Santos, K. Gummadi, and R. Rodrigues, "Towards Trusted Cloud Computing," Proc. HotCloud. June 2009.

[22] Wang, J., Shao, Y., Jiang, S., e Le, J. "Providing privacy preserving in cloud computing". Em International Conference on Test and Measurement, páginas 213–216. IEEE Computer Society. Hong Kong, 2009.

[23] Ghemawat, S., Gobioff, H., and Leung, Shun-Tak. The google file system. Proceedings of the nineteenth ACM symposium on Operating systems principles ACM. New York, Volume 37 Issue 5, December 2003. NY, USA.

[24] DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Sivasubramanian, S., Vosshall, P., and Vogels. "Dynamo: amazon's highly available key-value store". Proceedings of twenty-first ACM SIGOPS symposium on Operating systems principles. ACM. New York, NY, USA. 2007.

[25] Azure. Microsoft Azure. 2011.

[26] D. Borthakur, "The Hadoop Distributed File System: Architecture and Design". The Apache Software Foundation, 2007.

[27] Kai Hwang, Sameer Kulkareni, and Yue Hu, "Cloud Security with Virtualized Defense and Reputation-Based Trust Mangement," 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC), Chengdu, pp.717-722. China 2009.

[28] Rajkumar, B.. et al; Rajkumar Buyya, Rajiv Ranjan and Rodrigo N. Calheiros. (2009). "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", Proceedings of the 7th High Performance Computing and Simulation Conference (HPCS 2009, ISBN: 978-1-4244-4907-1, IEEE Press, New York, USA), Leipzig, Germany, June 21-24.

[29] Amazon; Amazon (2012). Amazon Web Services. Acessado em 01/06/2012. Disponível em: http://aws.amazon.com/pt/ec2/instance-types/.