IPHV8I2001X
International Journal Of Advanced Research and Innovation -Vol.8, Issue .II
ISSN Online: 2319 – 9253
Print: 2319 – 9245

TAS: ADAPTIVE TRUST MODEL FOR NETWORK INTEGRATION IN CLOUD

[#1]KUNA AKHILA, M.Tech Student,
[#2]S.NAVEEN KUMAR, Associate Professor,
Department Of CSE,
SREE CHAITANYA INSTITUTE OF TECHNOLOGICAL SCIENCES,KARIMNAGAR,T.S, INDIA.

ABSTRACT: The recent advances in cloud computing have risen a number of unforeseen security related issues in different aspects of cloud environments. Among these, the problem of guaranteeing secure access to computing resources in the cloud is gathering special attention. In this paper, we address open issues related to trust in cloud environments proposing a new trust model for cloud computing which considers a higher level view cloud resources. A simulation of trust calculation between the nodes of the clouds is performed. The simulation was possible to verify that a node is reliable when it reaches the minimum index of trust.

Keywords-*Cloud Computing; Distributed Computing; Security; Integrity; Confidentiality; Trust and Availability.*

I. INTRODUCTION

The widespread use of Internet connected systems and distributed applications have triggered a revolution towards the adoption of pervasive and ubiquitous cloud computing environments. These environments allow users and clients to purchase computing power according to necessity, elastically adapting to different performance needs while providing higher availability. Several web-based solutions, such as Google Docs and Customer Relationship Management (CRM) [2] applications, now operate in the software as a service model. Much of this flexibility is made possible by virtual computing methods, which can provide adaptive resources and infrastructure in order to support scalable on demand sales of such applications. Virtual computing is also applied to stand-alone infrastructure as a service solutions, such as Amazon Elastic Cloud Computing (EC2) and Elastic Utility Computing Architecture Linking Your Programs to Useful Systems (Eucalyptus) [2]. As a result, the cloud computing frameworks and environments are able to address different issues in current distributed and ubiquitous computing systems. The availability of infrastructure as a service and platform as a service environments provided a fundamental base for building cloud computing based applications. It also motivated the research and development of technologies to support new applications. As several large companies in the communications and information technology sector have adopted cloud computing based applications, this approach is becoming a de facto industry standard, being widely adopted by different organizations. Since the adoption of the cloud computing paradigm by IBM Corporation around the end of 2007, other companies such as Google (Google App Engine), Amazon (Amazon Web Services (AWS), EC2 (Elastic Compute Cloud) and S3 (Simple Storage Service)), Apple (iCloud) and Microsoft (Azure Services Platform) have progressively embraced it and introduced their own new products based on cloud computing technology [11]. However, cloud computing still poses risks related to data security in its different aspects (integrity, confidentiality and authenticity). Cloud computing provides a low-cost, scalable, location independent infrastructure for data management and storage. The rapid adoption of Cloud services is accompanied by increasing volumes of data stored at remote servers, so techniques for saving disk space and network bandwidth are needed. A central up and coming concept in this context is deduplication, where the server stores only a single copy of each file, regardless of how many clients asked to store that file. All clients that store the file merely use links to the single copy of the file stored at the server. Moreover, if the server already has a copy of the file, then clients do not even need to upload it again to the server, thus saving bandwidth as well as storage (this is termed client-side deduplication). Reportedly, business applications can achieve deduplication ratios from 1:10 to as much as 1:500, resulting in disk and bandwidth savings of more 90%. Deduplication can be applied at the file level or at the block level. In a typical storage system with deduplication, a client first sends to the server only a hash of the file and the server checks if that hash value already exists in its database. If the hash is not in the database then the server asks for the entire file. Otherwise, since the file already exists at the server (potentially uploaded by someone else), it tells the client that there is no need to send the file itself. Either way the server marks the client as an owner of that file, and from that point on the client can ask to restore the file (regardless of whether he was asked to upload the file or not). The client-side deduplication introduces new security problems. For example, a server telling a client that it need not send the file reveals that some other client has the exact same file, which could be sensitive information. A malicious client can use this information to check whether specific files were

uploaded by other users, or even run a brute force attack which identifies the contents of certain fields in files owned by other users, by trying to upload multiple variants of the same file which have different values for that field. The findings apply to popular file storage services such as Mozy Home and Drop box, among others. In this paper, we review the main cloud computing architecture patterns and identify the main issues related to security, privacy, trust and availability. In order to address such issues, we present a high level architecture for trust models in cloud computing environments. This paper is organized as follows. In Section II, we present an overview of cloud computing, presenting a summary of its main features, architectures and deployment models. In Section III, we present related works. In section IV, we introduce the proposed trust model. Finally, in Section V, 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 academia. 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" [14]. This definition includes cloud architectures, security, and deployment strategies. 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). Computational trust can be leveraged in order to establish an architecture and a monitoring system encompassing all these needs and still supporting usual activities such as planning, provisioning, scalability and security. Chang et al. [15] present a few challenges related to security, performance and availability in the cloud.

A. Characteristics of Cloud Computing

One advantage of cloud computing is the possibility of accessing applications directly from the Internet, with minor requirements of user computing resources. There are other significant advantages and disadvantages [13], as shown in Table I. Cloud computing combines a shared and statistical service model. It presents three basic characteristics [1]: a) hardware infrastructure architecture – based on low cost scalable clusters. The computing infrastructure in the cloud is composed of a great number of low cost servers, such as standard X86 server nodes; b) collaborative development of basic services and applications with maximal resource utilization, thus improving traditional software engineering processes. In the traditional computational model, applications become completely dependent on the basic services; c) the redundancy among several low cost servers is guaranteed through software. Since a large number of low cost servers is used, individual node failures cannot be ignored. Therefore, node fault tolerance must be taken into account in the design of software.

TABLE I. ADVANTAGES AND DISADVANTAGES OF CLOUD COMPUTING

Advantages	Disadvantages
Lower IT infrastructure cost	Requires a constant Network connection
Increased computing power	Dependable of network bandwidth
Unlimited storage capacity	Features might be limited
Improved compatibility between operating Systems	Stored data might not be secure
Easier group collaboration	If the cloud loses your data, you will not have access to your information.
Universal access to documents	

B. Cloud Computing Architecture

Cloud computing architecture is based on layers. Each layer deals with a particular aspect of making application 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. 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 [11].

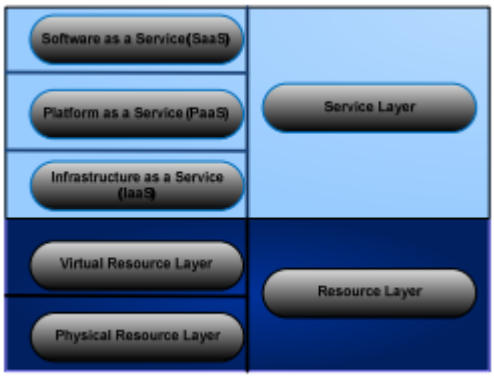

Figure 1. Cloud Computing Architecture [11]

C. Software as a Service

Software as a Service (SaaS) provides all the functions of a traditional application, but provides access to specific applications through Internet. The SaaS model reduces concerns with application servers, operating systems, storage, application development, etc. Hence, developers may focus on innovation, and not on infrastructure, leading to faster software systems development. SaaS systems reduce costs since no software licenses are required to access the applications. Instead, users access services on demand. Since the software is mostly Web based, SaaS allows better integration among the business units of a given organization or even among different software services. Examples of SaaS include [2]: Google Docs and Customer Relationship Management (CRM) services.

D. Platform as a Service

Platform as a Service (PaaS) is the middle component of the service layer in the cloud. It offers users software and services that do not require downloads or installations. PaaS provides an infrastructure with a high level of integration in order to implement and test cloud applications. The user does not manage the infrastructure (including network, servers, operating systems and storage), but he controls deployed applications and, possibly, their configurations [4]. PaaS provides an operating system, programming languages and application programming environments. Therefore, it enables more efficient software systems implementation, as it includes tools for development and collaboration among developers. From a business standpoint, PaaS allows users to take advantage of third party services, increasing the use of a support model in which users subscribe to IT services or receive problem resolution instructions through the Web. In such scenarios, the work and the responsibilities of company IT teams can be better managed. Examples of SaaS [2] include: Azure Services Platform (Azure), Force.com, EngineYard and Google App Engine.

E. Infrastructure as a Service

Infrastructure as a Service (IaaS) is the portion of the architecture responsible for providing the infrastructure necessary for PaaS and SaaS. Its main objective is to make resources such as servers, network and storage more readily accessible by including applications and operating systems. Thus, it offers basic infrastructure on-demand services. IaaS has a unique interface for infrastructure management, an Application Programming Interface (API) for interactions with hosts, switches, and routers, and the capability of adding new equipment in a simple and transparent manner. In general the, user does not manage the underlying hardware in the cloud infrastructure, but he controls the operating systems, storage and deployed applications. Eventually he can also select network components such as firewalls. The term IaaS refers to a computing infrastructure, based on virtualization techniques that can scale dynamically, increasing or reducing resources according to the needs of applications. The main benefit provided by IaaS is the payper-use business model [4]. Examples of IaaS [2] include: Amazon Elastic Cloud Computing (EC2) and Elastic Utility Computing Architecture Linking Your Programs To Useful Systems (Eucalyptus).

F. Roles in Cloud Computing

Roles define the responsibilities, access and profile of different users that are part of a cloud computing solution. Figure 2 presents these roles defined in the three service layers [3]. The provider is responsible for managing, monitoring and guaranteeing the availability of the entire structure of the cloud computing solution. It frees the developer and the final user from such responsibilities while providing services in the three layers of the architecture. Developers use the resources provided by IaaS and PaaS to provide software services for final users. This multi-role organization helps to define the actors (people who play the roles) in cloud computing environments. Such actors may play several roles at the same time according to need or interest. Only the provider supports all the service layers.

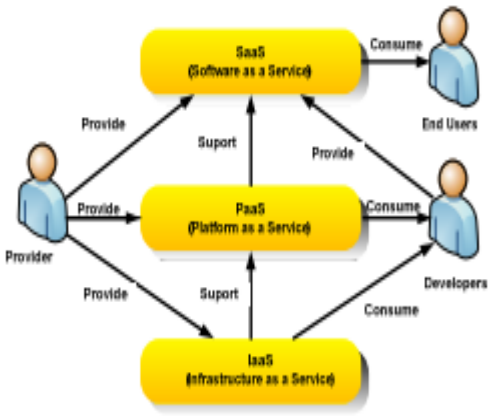

Figura 2. Roles in cloud computing [3].

G. Cloud Computing Deployment

According to the intended access methods and availability of cloud computing environments, there are different models of deployment [4]. 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, which are briefly analyzed below.

TABLE II. MODELS OF DEPLOYMENT OF CLOUD SERVICES [4]

Cloud Model	Description
Private	In this model, the cloud infrastructure is exclusively used by a specific organization. The cloud may be local or remote, and managed by the company itself or by a third party. There are policies for accessing cloud services. The techniques employed to enforce such private model may be implemented by means of network management, service provider configuration, authorization and authentication technologies or a combination of these
Public	Infrastructure is made available to the public at large and can be accessed by any user that knows the service location. In this model, no access restrictions can be applied and no authorization and authentication techniques can be used
Community	Several organizations may share the cloud services. These services are supported by a specific community with similar interests such as mission, security requirements and policies, or considerations about flexibility. A cloud environment operating according to this model may exist locally or remotely and is normally managed by a commission that represents the community or by a third party.
Hybrid	Involves the composition of two or more clouds. These can be private, community or public clouds which are linked by a proprietary or standard technology that provides portability of data and applications among the composing clouds.

Private Cloud computing presents a few challenges related to protection, trust, privacy and security of user data.

III. CLOUD RELATED WORK ON SECURITY AND TRUST

This section review some related work about security, file system and trust in the cloud.

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 [21]. 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 [7]. 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 as 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 [7]. Huan et al. [22] 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 compromised systems. Experimental results can be used to analyze the risk in third party compute clouds. Popovic et al. [23] 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. File system Security

As the number of devices managed by users is continually increasing, there is a growing necessity of synchronizing several hierarchically distributed file systems using ad-hoc connectivity. Uppoor et al. [6] present a new approach for synchronizing of hierarchically distributed file systems. Their approach resembles the advantages of peerto-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 [9] 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. An implementation of the proposed scheme using an open source distributed file system named HDFS (Hadoop Distributed File System) is discussed. Experimental measurements point out that the dynamic scheme outperforms existing static file distribution algorithms.

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 in 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 [5]. 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. When a virtual machine is started or a migration takes place, the TC verifies whether the node is trustworthy so that the user of the virtual machine may be sure that the platform remains trustworthy. A key and a signature are used for identifying the node. In the TCCP model, the private certification authority is involved in each transaction together with the TC [5]. Shen et al. [7] 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 [7]. 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. [8] introduced a multitenancy 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. [12] 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 is divided in intra and interdomain 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. If a node needs to calculate the trust value of another node, it first checks the direct trust table and uses that value if the value corresponding to the desired node is already available. Otherwise, if this value is not locally available, the requesting node checks the recommendation list in order to determine a node that has a direct trust table that includes the desired node. Then it checks the direct trust table of the recommended node for the trust value of the desired node. The process continues until a trust value for the desired node is found in a direct trust table of some node. The interdomain trust values are calculated based on the transactions among the inter-domain nodes. The inter-domain trust value is a global value of the nodes direct trust values and the recommended trust value from other domains. Two tables are maintained in the Trust Agents deployed in each domain: form of Inter-domain trust relationships and the weight value table of this domain node. In [17] 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 user's VM, and allows a user to determine up front whether or not the IaaS enforces these properties. The work [18] 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.

V. CONCLUSION

We have presented an overview of the cloud computing paradigm, as well as its main features, architectures and deployment models. Moreover, we identified the main issues related to trust and security in cloud computing environments. In order to address these issues, we proposed a trust model to ensure reliable exchange of files among cloud users in public clouds. In our model, the trust value of a given node is obtained from a pool of simple parameters related to its suitability for performing storage operations. Nodes with greater trust values are subsequently chosen for further file storage operations. As a future work, we plan to

implement the proposed trust model and analyze node behavior after the ranking of trustworthy nodes is established.

REFERENCES

[1] Chen Kang and Zen WeiMing, "Cloud computing: system instance and current research," Journal of Software, pp. 20(5):1337-1347. 2009.

[2] 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).

[3] 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).

[4] Zhao-xiong Zhou, He Xu, and Suo-ping Wang, "A Novel Weighted Trust Model based on Cloud," AISS: Advances in Information Science and Service Sciences, Vol. 3, No. 3, pp. 115- 124, April 2011.

[5] 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.

[6] S. Uppoor, 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.

[7] 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.

[8] 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.

[9] Qingsong Wei, Bharadwaj Veeravalli, Bozhao Gong, Lingfang Zeng, and Dan Feng, "CDRM: A Cost-Effective Dynamic Replication Management Scheme for Cloud Storage Cluster," 2009 IEEE International Conference on Cluster Computing (CLUSTER), pp. 188-196, 2010.

[10] 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 '09), pp. 717-722, 2009.