2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks

Determining Service Trustworthiness in Intercloud Computing Environments

Jemal Abawajy
Deakin University
School of Information Technology
jemal@deakin.edu.au

***Abstract*—Deployment of applications and scientific workflows that require resources from multiple distributed platforms are fuelling the federation of autonomous Clouds to create Cyberinfrastructure environments. As the scope of federated cloud computing enlarges to ubiquitous and pervasive computing, there will be a need to assess and maintain the trustworthiness of the cloud computing entities. In this paper, we present a fully distributed framework that enable interested parties determine the trustworthiness of federated cloud computing entities.**

***Index Terms*— Cloud computing, Grid computing, reputation, trust, interGrid.**

1.0 Introduction

Advances in systems such as hardware, networking, middleware and increasing ubiquity of Virtual Machine (VM) technologies have lead to an emergence of new globally distributed computing platforms such as Grid computing [8, 9, 10, 11, 12] and cloud computing [16, 17] that provides software, computing resources and storage as a service for fee accessible from anywhere via the Internet. Although these new generations distributed computing platforms have been used for various applications, they generally follow specific requirements of their user communities.

In this paper, we refer to Grid-based distributed computing as public cloud computing whereas distributed computing provided by Google, Amazon, Microsoft and others that allows workloads to be deployed and scaled-out quickly through the rapid provisioning of virtual machines or physical machines as private cloud computing. For example, Amazon's Elastic Compute Cloud (EC2) [16] allows users to deploy VMs on demand on Amazon's infrastructure and pay only for the computing, storage and network resources they use.

Generally, existing clouds are specific to each owner and unaware of the existence of other clouds. As a result, there is hardly any resource and service sharing between them. In this paper, we refer to the logical federation of autonomous computing clouds for the purpose of exchanging resources (storage, compute, messaging etc) in a uniform/unified way as inter-cloud computing.

There are ample benefits for interconnecting computing clouds in a uniform way while respecting their autonomy. For example, the federated clouds will enable users to solve large-scale computational and data intensive problems in science, engineering, and commerce. These benefits have inspired research in creating mechanisms and protocols for interlinking exiting Grids across multi-site in a coordinated manner [2, 12, 18].

Although there are values for federating autonomous clouds, the open and dynamic nature of these systems coupled with the independent capacity planning and provisioning of resources to users within each system makes resource sharing in inter-cloud computing environment a challenging task. Issues such as standardization of network protocols and the mechanism that would allow them to interwork such as the interfaces through which cloud systems internetwork with each other as well as enabling the provisioning of reliable cloud services are needed to fully realize inter-cloud computing. Specifically, since inter-cloud computing constitutes collaboration between independently owned autonomous clouds, there is a need for policies and mechanisms for these clouds to peer with each other and for admission control when accepting requests originated from other clouds. We also need mechanisms for selecting trustworthy clouds to peer with and outsource applications for execution or date for storage.

In this paper, we focus on the problem of how to determine service trustworthiness in inter-cloud computing environments. Policies and mechanisms for peering Grids and for admission control have been discussed in [2]. Although, it has been clearly shown that an assurance of a higher degree of trust relationship is required to attain efficient resource allocation and utilization [6], to the best of our knowledge, the problem of how to determine service trustworthiness in inter-cloud computing environments has not been addressed previously.

In this paper, we present a fully distributed framework that enables interested parties to determine the trustworthiness of inter-cloud computing entities. The proposed trust framework is a reputation-based trust management system that enables a service requester to obtain service trustworthiness. The proposed trust management framework model enables users to select high-quality cloud services through determining the trustworthiness of a given resource for the purpose of executing their jobs, thereby satisfying clients' quality-of-service (QoS) requirements.

The rest of the paper is organised as follows. The background and related work are discussed in Section 2. We also discuss the problem of trustworthy resource selection and provisioning in inter-cloud computing environments. In Section 3, the architecture of the Inter-cloud computing and the proposed trust framework model are discussed. We discuss the representation of reputation and how the reputation is built. We also discuss how reputation is updated as well as how the ratings of others are considered and integrated. In Section 4,

978-0-7695-3908-9/09 $26.00 © 2009 IEEE
DOI 10.1109/I-SPAN.2009.155

the analysis of the proposed algorithm is given. The conclusions and future directions are explained in Section 5.

2.0 Background and Related Works

Scientists and practitioners need a large-scale distributed computing system to cope with the scale and complexity of both current and next generation scientific challenges. The aim of the inter-cloud computing is to enable the creation of a virtual Cyberinfrastructure computing environment. Inter-cloud computing allows users to share computational resource, storage resource, networks, application services, or other types of resources. The resource sharing could be based on provisioning policies that are determined by the service providers' perception of utility. For example, the owner of a cloud could allocate a share of the resources in return for regular payments.

Recent efforts have demonstrated interest in resource sharing across multi-site networks, such as Grids, in a coordinated manner. PlanetLab architecture is evolving to be deployed by other organisations and enable federations of PlanetLab's [19]. Similarly, the Grid'5000 comprises nine sites geographically distributed in France [12]. An architecture and mechanisms based on the idea of peering arrangements between Grids to enable resource sharing across Grids is described in [2]. The focus of these prior works has been on the problem of resource exchange between different Grids. In this paper, we extend the interGrid architecture to enable inter-cloud computing users to determine the trustworthiness of a Grid for the purpose of outsourcing application and data for execution.

However, collaboration is only productive if all participants operate in an honest manner and, therefore, establishing and quantifying trust, which is the driving force for collaboration, is important for realising the benefits of inter-cloud computing. Trust is the firm belief in the competence of an entity to act as expected within a specific context at a given time [13]. Reputation is a measure that is derived from direct or indirect knowledge of earlier interactions of peers and is used to access the level of trust a peer puts into another [3, 13]. As an entity can trust another entity in the network based on a good reputation, we can use reputation to build trust [7]. This means reputation can serve, in the sense of reliability, as a measure of trustworthiness.

2.1 Problem Statement

In inter-cloud computing, users and computational agents and services often interact with each other without having sufficient assurances about the behavior of the resource they entrust their data and applications with. There is often insufficient information for deciding which resources to use. As the scope of inter-cloud computing enlarges to ubiquitous and pervasive computing, there will be a need to assess and maintain the reputation of the entities.

It is necessary to create a reputation manager that could capture and efficiently store the behaviour of entities, while being able to update it with new information if possible. A reputation system should have efficient representation of reputation as well as efficient mechanism for updating reputation and integrating efficiently the ratings of others. Another key problem associated with the formation and operation of inter-cloud computing is that of what kind of information to collect and how to specify and enforce community trust. Also, distributing the reputation information about other cloud providers is an extremely important property of trust management systems. However, subverted clouds may lie and misreport about the service quality they received from a given service providers. Thus, distributing the reputation information must be handled carefully.

2.2 Related work

In this paper, we explore the potential of reputation management mechanisms that are based on some aspects of social control. Trust and reputation systems have been recognized as playing an important role in decision making in the Internet world [7]. The recent work on trust management for Grid computing shows that modeling trust is of great importance for the future developments of the Grid computing [6]. As a result, integration of trust management system in standard grid computing has lately received attention [3, 6, 13]. For example, a trust brokering system that operates in a peer-to-peer manner is proposed in [13]. An extension of Grid information service with reputation management service and its underlying algorithm for computing and managing reputation in service-oriented grid computing is discussed in [6].

Although exiting works are complementary to the work proposed in this paper, to the best of our knowledge, this is the first work that attempts to build trustworthiness in inter-cloud computing. Exiting reputation systems for the standard Grids also suffer from a number of attacks that weaken trust management systems. The proposed trust model prevents many of such attacks and improves the reliability and the welfare of the system.

3.0 Determining Service Trustworthiness

In this section, we present a brief discussion of the main components of the inter-cloud computing architecture and the proposed mechanism for determining trustworthiness of a given resource.

3.1 System Architecture

Let $G_1, G_2, G_3, \ldots$ be the universe of autonomous clouds. By autonomous we mean that no cloud has direct control and power over the actions of another cloud. For the purposes of this paper, we define a federated cloud computing as a subset of the universe of clouds consisting of $G = (G_1, G_2, \ldots, G_n)$.

Figure 1 shows the basic components of the inter-cloud computing infrastructure that allows users to deploy applications and scientific workflows that require resources beyond the capacity of their clouds. The architecture is layered in that services are provisioned to cloud users based on cloud-level or inter-cloud level by two resource management policies.

Cloud users require resources to deploy services and run applications. Cloud providers provide resources and services to potential users for fee or following another economic model

such as bartering. Resource providers have their cost structures and policies that govern how their resources are provisioned to a user.

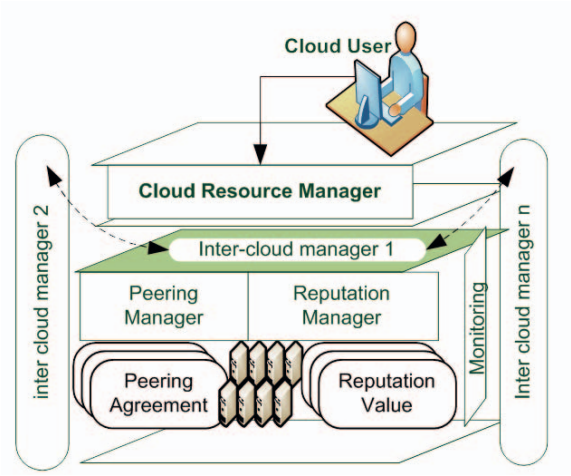

Figure 1: InterCloud computing infrastructure

We assume that the conditions under which the resources are provided to the users are stated in contracts such as Service Level Agreements (SLAs). We also assume that the transacting nodes will have SLA contract prior to starting the service. The SLA contract is also digitally signed by both parties.

A. Peering Manager

Resources sharing between multiple clouds to meet cloud user requirements are enabled by peering arrangements established between the participating clouds. This is briefly described in this subsection.

Cloud providers acquire shares from one another following peering policy to meet the needs of their client demands. A Cloud $G_i \in G$ provider can have peering arrangements with another cloud $G_j \in G$ provider through which they coordinate sharing of the use of resources of the inter-cloud computing. The peering agreement describes the information that is to be exchanged under the terms stated in contracts such as SLA. The peering policy also describes the desired level of access control as well as mechanisms to protect data both in storage and transmission.

B. Grid Resource Manager

Users submit their resource allocation requests to their local Cloud Resource Manager (CRM). The CRM is responsible for the resource provisioning and allocation at the individual cloud level. If CRM cannot fulfill the user request locally or the application requires resources from multiple clouds, CRM forwards the request to the inter-cloud resource manager (IRM) discussed in the next subsection.

C. Inter-cloud Resource Manager

The functionality of the IRM is similar to the InterGrid Gateway (IGG) discussed in [2] but extended with a number of capabilities. The IRM is responsible to mediate the resource exchange between peering clouds. A given IRM is responsible for establishing terms of the peering agreements with other clouds and the types of resources that the peering clouds can acquire from one another.

IRM also provides cloud selection capabilities by selecting suitable clouds that are able to provide the required resources to users' requests. It is also responsible for managing requests for resources and services from other cloud IRMs.

IRM is also responsible for monitoring the execution of applications across multiple clouds. The IRM also interacts with other entities including accounting systems that provide information on shares consumed by peering clouds.

D. Reputation manager

The reputation manager is in charge of collecting, calculating and maintaining a certain trust measurement of peering clouds. The rating is based on direct observation and experience (i.e., first-hand information) and indirectly by sharing observations and experience measures with other entities (i.e., second-hand information).

Let $A \subseteq G = \{C_1, C_2, \ldots C_m\}$ be a set of autonomous clouds with a peering agreement. The clouds in $A \subseteq G$ can share computational resource, storage resource, networks, application services, or other types of resources for fee. They are also given the ability to exchange first-hand service satisfaction about a given service provider with each other. The reputation manager can use this information to infer and store the reputation of the members of its peering. Such reputation will be later used by IRM to obtain the trust values that can be used for the purpose of deciding the selection of the best partner for a certain operation, or discover if one entity is behaving maliciously.

3.2Reputation Representation

In this paper, we specify trust in the form of a trust relationship between two peering entities. This trust is always related to a particular context. Each cloud maintains and computes its reputation locally. T The reputation manager uses the following ratings for building the reputation of an entity:

- *personal experiences* ($P_{i,j}$) represents the *first hand information of* $G_i \in A$ after transacting with Grid $G_j \in A$.
- *reputation rating* ($R_{i,j}$) represents the confidence $G_i \in A$ formed $G_j \in A$'s behavior as a good service provider.
- *honesty rating* ($H_{i,j}$) represents $G_i \in A$'s opinion about how credible $G_j \in A$ is as a provider of second-hand information.

The reputation manager is entrusted with the task of collecting and maintaining reputation rating, honesty rating and personal experience rating (i.e., first-hand information) about every other cloud provider that it has peering arrangement with. The reputation rating ($R_{i,j}$) and honesty rating ($H_{i,j}$) are maintained privately while personal experiences ($P_{i,j}$) can be shared with the members of the peering group. In the proposed approach, entities can only share first-hand information. For example, a cloud $G_i \in A$ will only share with a cloud $G_k \in A$ its first hand experience about cloud $G_j \in A$. This allows us to avoid dependence and loops of reputation ratings. An IRM uses reputation rating ($R_{i,j}$) and honesty rating ($H_{i,j}$) to periodically classify other nodes as trustworthy/untrustworthy.

3.3Reputation update algorithm

Once the rating (i.e., first-hand or second-hand) information has been gathered, the reputation manager can update the reputation records.

Fig. 2 presents the procedure for updating reputation records. The update algorithm is triggered whenever $G_i \in A$ experiences first-hand ($P_{i,j}$) interaction with $G_j \in A$ or $G_i \in A$ receives second-hand rating ($P_{k,j}$) from $G_k \in A$ about $G_j \in A$ or after certain period is elapsed.

```
1. Algorithm UpdateTrust
2. INPUT: G_j ∈ A, P_x,j
3. BEGIN
4.   IF (P_x,j | x == i) THEN
5.     Update first hand information
6.     Update reputation rating
7.     Update honesty rating
8.   ELSE IF (P_x,j | x == k) THEN
9.     IF (sourced from peers) THEN
10.      Update Gj reputation rating based on Quorum
11.      Update Gj honesty rating based on Quorum
12.    ELSEIF (honesty (G_k ∈ A) == true) THEN
13.      Update reputation rating about Gj
14.    ELSEIF (viewTrust (P_k,j) ≥ honestThreshold) THEN
15.      Update reputation rating about Gj
16.    ELSE
17.      FOR all G_i that rate G_k ∈ A honest THEN
18.        Update honesty rating of G_i
19.      ENDFOR
20.    ENDIF
21.      Update trust rating about Gj
22.  ENDIF
23. END UpdateTrust
```

Figure 2: Trust update algorithm

We assume that, from time to time, Grids publish their personal experiences ($P_{i,j}$) in the form of rating to a subset of their peers that they have a peering agreement with. Another case where the second-hand information used is when a Grid $G_i \in A$ does not have a reputation rating for a Grid $G_j \in A$ in its database or it has purged it. Reputation is built over time, using the behaviour of the nodes as a feedback. Due to the possible existence of subverted clouds, a trust entity for clouds faces the problem of integrating dishonest ratings. Thus, the challenge is how to systematically incorporate second-hand information collected from other clouds into computing trustworthiness of a given cloud.

Service providers entering into a new peering agreement may source the trustworthiness information from the peers that it has agreements with. To take advantage of the sourced reputation information (i.e., to learn from observations made by others before having to learn by own experience), we use a quorum to update both the reputation and credibility ratings of the node.

To handle subverted clouds that may lie and misreport about the service quality they received from a given service providers, each cloud maintains an honesty level threshold for deciding whether or not to consider the second-hand rating from cloud $G_k \in A$. For example, a cloud that reports inconsistent values cannot be trusted as a source of true second-hand information data. Also, we empower each cloud to reduce the honesty level of those clouds that support dishonest cloud. Specifically, let $G_i \in A$ receives unsatisfactory services from $G_k \in A$. In this case, $G_i \in A$ will then reduces the honesty ratings of all those who praised $G_k \in A$ as good service provider. This will minimize the collusion problem.

4.0 Analysis of the Algorithm

A federated system composed of autonomous distributed systems can pose several risks to the user communities. Reputation management has an important role in establishing cooperative relationships between users and service providers by lowering some of these risks [3]. Trust can be used to measure our confidence that a secure system behaves as expected. A reliable trust management system provides capability to convert the unpredictable, highly dynamic pervasive environment into a trusted business platform. Therefore, reliability of the trust management is one of the important metrics to analyse the strength of a given trust management system. A reliable trust management system should help the users to defend themselves against malicious information, including trust values propagated by other users into the system. The system is reliable if this property is accomplished.

In this paper, we presented a reputation manager that captures and efficiently stores the behaviour of other entities in the previous interactions, while being able to update it with new reputation information. The advantages of the proposed scheme are that it provides a means for good entities to avoid working with less trustworthy parties. Malicious users, whose behavior has caused them to be recognised as having low trustworthiness, will have less ability to interfere with network operations.

Even though inter-cloud resource manager have imprecise information about their peered clouds, the proposed approach enables inter-cloud resource manager to determine the trustworthiness level of a cooperating cloud. This will help decide which clouds to get into peering arrangement or outsource application exaction or data storage to. Moreover, the results of trust evaluation can be directly applied to detect selfish and malicious entities in the network.

The proposed reputation management system is reliable as it helps clouds to defend themselves against malicious information, including rating values propagated by other clouds into the system. The proposed reputation-based system is reliable as it minimises the dissemination of dishonest ratings by the peering clouds at various levels. Note that the second-hand information is the rating resulted from the bilateral direct interactions between the trustor and the trustee. Nodes without prior direct interactions cannot publish this information. This is because when a cloud $G_i \in A$ deliberates to enter a transaction with $G_j \in A$, the two clouds complete and

digitally sign an SLA contract. This ensures that both the service provider and the service user cannot deny entering into contractual agreement.

The proposed trust management framework model enables users to select high-quality cloud services through determining the trustworthiness of a given resource for the purpose of executing their jobs, thereby satisfying clients' quality-of-service (QoS) requirements.

5.0 Conclusion

A number of distributed computing models with aims to provide reliable, customized and QoS guaranteed computing dynamic environments for end-users have recently emerged. These distributed computing platforms generally follow specific requirements of their user communities and as a result mostly work in isolation with little or no resource sharing between them. To remedy this situation, several inititatives that conglomerate autonomous clouds into large-scale distributed computing platforms have recently been underway. In such systems, however, users and computational agents and services often interact with each other without having sufficient assurances about the behavior of the other party.

In this paper, we presented a distributed reputation-based trust management system for inter-cloud computing system. Trust value storage is distributed at the levels of the clouds in the system, which enables each cloud to make independent local decision for selection about trustworthiness of a cloud. We have studied the performance of the proposed trust management system in a simulated environment and due to space limitations this information is not provided here. Trust management is obviously an attractive target for adversaries. Besides well-known straightforward attacks such as providing dishonest ratings, some sophisticated attacks can undermine the whole trust evaluation process. We are currently developing a full list of threats against the proposed trust management and analysing the vulnerability of the system to these threats.

Acknowledgement: This research is supported by ARC Discovery grant. The help of Maliha Omar, without whom this work will be difficult, is also appreciated.

6.0 References

[1]. Pinyol, I., Paolucci, M., Sabater-Mir, J., Conte, R.: Beyond accuracy. Reputation for partner selection with lies and retaliation. In: MABS, Honolulu, 15 May 2007

[2]. Marcos Dias de Assuncao: Provisioning Techniques and Policies to Enable Inter-Grid Resource Sharing, PhD Thesis, Melbourne University, 2009.

[3]. Jemal H. Abawajy, Andrzej M. Goscinski: A Reputation-Based Grid Information Service. International Conference on Computational Science (4) 2006: 1015-1022.

[4]. Anthony Davison. Statistical Models. Cambridge University Press, Cambridge Series in Statistical and Probabilistic Mathematics, ISBN: 0521773393, October 2003.

[5]. S. Buchegger and J. Le Boudec, "A robust reputation system for p2p and mobile ad-hoc networks," Proceedings of the 2nd Workshop on Economics of Peer-to-Peer Systems, 2004.

[6]. V.Vijayakumar and R.S.D.Wahida Banu, International Journal of Computer Science and Network Security, VOL.8 No.11, pp, 107-115, 2008.

[7]. A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. Decision Support Systems, 43(2):618–644, March 2007.

[8]. K. Miura, "Overview of japanese science grid project naregi," Progress in Informatics, pp. 67–75, 2006.

[9]. "Open Science Grid," http://www.opensciencegrid.org, 2005.

[10]. L. Peterson, S. Muir, T. Roscoe, and A. Klingaman, "Planetlab architecture: An overview," PlanetLab Consortium, Princeton, USA, Tech. Rep. PDN-06-031, May 2006.

[11]. C. Catlett, P. Beckman, D. Skow, and I. Foster, "Creating and operating national-scale cyberinfrastructure services," Cyberinfrastructure Technology Watch Quarterly, vol. 2, no. 2, pp. 2–10, May 2006.

[12]. F. Cappello, E. Caron, M. Dayde, F. Desprez, Y. Jegou, P. Primet, E. Jeannot, S. Lanteri, J. Leduc, N. Melab, G. Mornet, R. Namyst, B. Quetier, and O. Richard. "Grid'5000: a large scale and highly reconRgurable grid experimental testbed". In 6th IEEE/ACM International Workshop on Grid Computing, 2005.

[13]. F. Azzedin and M. Maheswaran, " A Trust Brokering System and Its Application to Resource Management in Public Resource Grids", in Proceedings of IPDPS 2004.

[14]. L. Peterson and J. Wroclawski. Overview of the GENI architecture. GENI Design Document GDD- 06-11, GENI: Global Environment for Network Innovations, January 2007.

[15]. Grid Interoperability Now Community Group (GIN-CG). http://forge.ogf.org/sf/projects/gin, 2006.

[16]. Amazon Elastic Compute Cloud [URL]. http://aws.amazon.com/ec2, access on Oct. 2009.

[17]. IBM Blue Cloud project [URL]. http://www-03.ibm.com/press/us/en/pressrelease/22613.wss/, access on Oct 2009.

[18]. Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic, Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems, Volume 25, Number 6, Pages: 599-616, ISSN: 0167-739X, Elsevier Science, Amsterdam, The Netherlands, June 2009.

[19]. L. Peterson and J. Wroclawski, "Overview of the GENI architecture," GENI: Global Environment for Network Innovations, GENI Design Document GDD-06-11, January 2007. [Online]. Available: http://www.geni.net/GDD/GDD-06-11.pdf