A Holistic Trust Management Leasing Algorithm for IaaS Cloud

Hemant Kumar Mehta, School of Computer Science and IT, Devi Ahilya University, Indore, India
Rohit Ahuja, School of Computer Science and IT, Devi Ahilya University, Indore, India

ABSTRACT

Trust is an important factor in the exchange of services among multiple parties for example in cloud environment a large number of users interact in the form of resource provider and resource consumers. The resource consumer requests for computation service from the resource provider which provide service to the resource consumer. In this paper we proposed a trust management architecture that keeps track of past performance of resource provider and resource consumer so that every time the participating entity in a transaction has idea of behavior of other entities. This architecture not relies only on trust value of a resource provider, however it also considers several other parameters viz. activeness, ratio of positive and negative feedback etc. As the trust value of a resource provider is actually the result of user’s feedback which declines over time, hence we have also given chance of regret to the resource providers which were proven to be untrusted to convert their “setback to comeback”.

Keywords: Cloud Computing, Feedback, IaaS Cloud, Leasing Algorithms, Regret Management, Trust Based Leasing Algorithms

INTRODUCTION

Distributed computing has been evolved over a large span of time and various forms of distributed computing are under the active use. Cluster, Grid and Cloud computing are among most widely used distributed computing environments (DCEs). These DCEs provide computation facilities to their users, In cluster computing the workstations are loosely connected at the underlying architecture, Cluster Computing is followed by the Grid Computing, Grid is same as the traditional cluster computing the main difference is that grids are more loosely coupled, heterogeneous and geographically dispersed. Grid computing is followed by Cloud computing, the term cloud is used to hide the complex infrastructure details. Grid computing provides service to user by the computers in the same network whereas in cloud computing service is provided by the third party. Cloud Computing provides three types of service software as a service (SaaS), platform as a service (PaaS) and Infrastructure as a service (IaaS). All these three types of cloud are aimed to provide computing

DOI: 10.4018/ijssoe.2014040101
facility to the user in the form of services on the basis of pay as you go model.

IaaS cloud provides hardware resources i.e. processor, storage and network to its consumers. Consumers in turn pay for the resources used.

OpenNebula is an open source tool which is used for managing infrastructure of heterogeneous and distributed data center. OpenNebula is supports all four deployment models of IaaS cloud. Haizea is open source lease manager that can be integrated with OpenNebula. Haizea acts as a resource manager and used to schedule a lease to its desired resource. The most basic unit of Haizea is the lease. User submits its demand for resource in the form of a lease. Generally, there are following three types of lease format in which user can make request:

1. **Best-Effort (BE) Lease**: In BE lease resources are allocated as and when they are available;
2. **Advance Reservation (AR) Lease**: In AR lease resources are provisioned during a strictly defined time period;
3. **Immediate Lease**: In Immediate lease resources must be provisioned just after their submission (Sotomayor, 2010).

A number of users have been involved in cloud computing, hence a strong possibility exists that some of the users may start malicious behavior by providing wrong information. For example there may be a claim that the provider is having a workstation of higher capacity than the actual capacity to attract more consumers or the resource requirement mentioned in the lease is less than the actual use at run time to save the cost of execution. We call such providers and consumers as malicious. The wrong information provided by such malicious participants ultimately affects the system’s performance.

This paper proposes a trust management architecture that uses the concept of reputation points to ensure the trustworthiness of resource providers. Every lease comes with its reputation points, if the lease execution is successful then these reputation points will be added to provider’s trust value. Similarly in case of unsuccessful execution these reputation points will be deducted from provider’s trust value. Before submitting the lease the trust value of each provider will be referred and provider having highest trust value among the competent and capable providers will be selected. Besides the trust value we have also considered three more parameters activeness, ratio of positive & negative feedbacks and option for regret to have an effective trust management mechanism as explained in section 3. User’s feedback will not be just added or subtracted as a bonus or penalty after successful or unsuccessful execution of the process, to avoid user biasness the user’s feedback will be passed from the feedback verifier that checks the user feedback and corrects it if necessary.

**LITERATURE REVIEW**

Trust management is applied to a wide range of applications like security, social computing, multiparty computing etc. The focus of this paper is resource management algorithms, hence we limit the scope of this literature survey to the research work in the direction of scheduling algorithms using trust management.

Manuel et al. has proposed a system which chooses appropriate grid resource in heterogeneous environment based on the requirement of user. Feedback evaluator module is integrated which collects the user feedback and there is a feedback verifier module which maintains a criteria to verify whether the feedback given by the user is correct or incorrect or how much it is deviated from the threshold, if the feedback is less or more than a particular factor then it is rejected and then the feedback updater module receives the feedback from the feedback verifier module and sent to the feedback repository (Manuel, 2002).

Etalle et al. has enforced the concept of Regret and Forgiveness by the quote of Gandhiji “If we practice and eye for an eye and a tooth for a tooth, soon the whole world will be blind and toothless”. Trust is only meaningful when
Model Expansion in Model-Driven Architectures
www.igi-global.com/chapter/model-expansion-model-driven-architectures/65166?camid=4v1a

Translating Schemas Between Data Modelling Languages
www.igi-global.com/chapter/translating-schemas-between-data-modelling/23409?camid=4v1a