Introduction

Cloud infrastructure is complex and heterogeneous in nature, with numerous components provided by different vendors. Applications deployed in the cloud might need to interact amongst themselves and, in some cases, depend on other deployed applications.

The complexity of the cloud infrastructure and application dependencies create an environment which requires careful management and raises security and privacy concerns. Such management is beyond the capabilities of human management efforts and current management tools. It rather requires establishing trustworthy self-managed services.

The central component that manages the allocation of virtual resources of a cloud infrastructure's physical resources is known as the cloud scheduler [1,3]. Currently available schedulers do not consider users' security and privacy requirements, neither do they consider the properties of the entire cloud infrastructure. For example, a cloud scheduler should consider the application's performance requirements and users security and privacy requirements. We introduce a *trustworthy cloud scheduler* which we call Access-Control-as-a-Service (ACaaS).

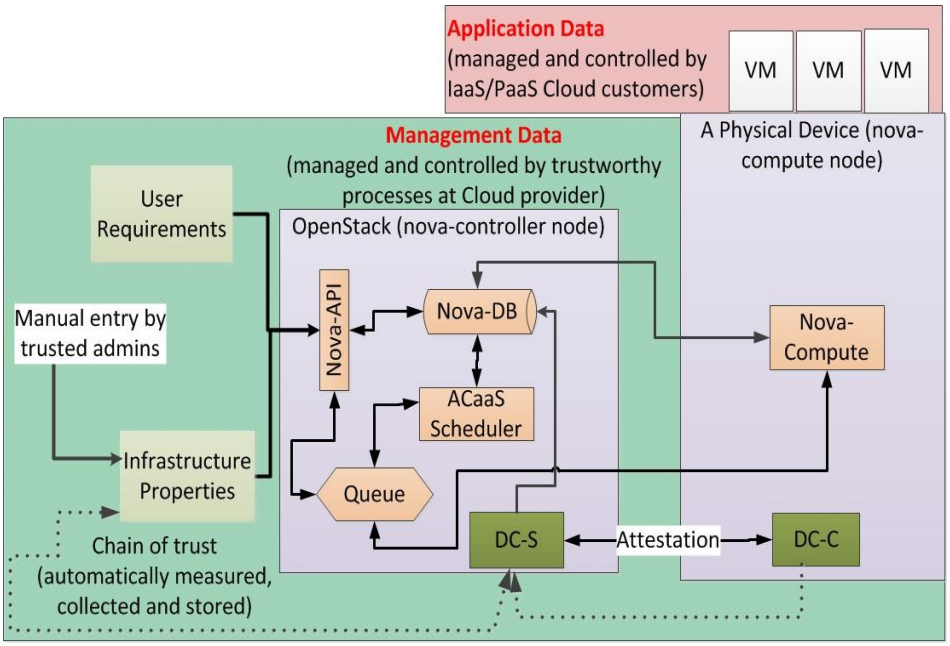

Figure 1: High level architect for ACaaS integration in OpenStack

High-level architecture and workflow

ACaaS is a novel cloud scheduler which considers both user requirements and infrastructure properties. It focuses on assuring users that their virtual resources are hosted using physical resources that match their properties without getting users involved with understanding the details of the complex cloud infrastructure.

As a proof-of-concept ACaaS has been prototyped and integrated as part of OpenStack. The provided prototype is composed of the following main components:

- a trusted scheduler,
- a server-side software agent (DC-S), and
- a client-side software agent (DC-C).

The DC-C runs on the physical cloud device and is in charge of enforcing the cloud policies. The DC-S, on the other hand, runs as part of OpenStack for managing the policies of the cloud and ensuring their execution. As part of this, DC-S establishes a chain-of-trust with each DC-C, collects the trust properties of the DC-C's physical resource, and passes the result to a central database. Whenever a physical resource fails, or a user submits a new service request, ACaaS is instructed to allocate a physical resource to host the new user request, or to move virtual resources from the failed physical resource to an appropriate physical resource. ACaaS in all cases considers both user requirements and the infrastructure properties.

Compositioal Chains of trust

Assessing the trust levels of a cloud is a difficult problem to deal with considering the dynamic nature and enormous resources of the infrastructure [3]. We propose the concept of compositional chains of trust which provides a single chain of trust representing a group of entities [3]. This is important in clouds as many entities exist as a composition of multiple entities (e.g. a cluster of physical servers, a cluster of load balanced application and database servers). Members of such a grouping may have identical or different chains of trust. However, to a client that depends on this grouping, they should appear as a single chain of trust representing the grouping. In other words, a client will see a single entity, even though that entity will be a grouping representing multiple entities. Moreover, the functions proposed to calculate the compositional chains of trust provide different levels of transparency based on the cloud resource type.

ACaaS uses the compositional chain of trust method to measure the trustworthiness of the cloud infrastructure. DC-C measures the trust level of each node and then pushes it to DC-S. DC-S stores the result in a database which is used by ACaaS.

Nova-API

Nova-API is used by customers when managing their resources in the cloud, and is also used by cloud administrators when managing the cloud virtual infrastructure. We introduced the following changes to nova-API:

i) enable users to manage their requirements;
ii) enable administrators to manage the properties and policies of the infrastructure;
iii) enable an automated collection of the physical resources' properties through trustworthy channels.

These data are used by the ACaaS scheduler.

References

[1] I. M. Abbadi and A. Ruan. Towards Trustworthy Resource Scheduling in Clouds (to appear). In the IEEE Transactions on Information Forensics & Security, 2013.

[2] I. M. Abbadi and C. Namiluko. Dynamics of Trust in Clouds — Challenges and Research Agenda. In Proc. 6th International Conference for Internet Technology and Secured Transactions (ICITST), IEEE, 2011.

[3] I. M. Abbadi. Clouds Trust Anchors. In Proc. 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom−11), IEEE, 2012.

Further Information

Further information about Access-Control-as-a-Service can be found in Deliverable „D2.3.2—Components and Architecture of Security Configuration and Privacy Management".

Disclaimer

The TClouds project has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement number ICT-257243.

TClouds at a glance

Project number:
257243

TClouds mission:
- Develop an advanced cloud infrastructure that delivers computing and storage with a new level of security, privacy, and resilience.
- Change the perceptions of cloud computing by demonstrating the prototype infrastructure in socially significant application areas.

Project start:
01.10.2010

Project duration:
3 years

Total costs:
EUR 10.536.129

EC contribution:
EUR 7.500.000

Consortium:
14 partners from 7 different countries.

Project Coordinator:
Dr. Klaus-Michael Koch
coordination@tclouds-project.eu

Technical Leader:
Dr. Christian Cachin
cca@zurich.ibm.com

Project website:
www.tclouds-project.eu