JOURNAL OF INFORMATION SECURITY AND APPLICATIONS 27-28 (2016) 65–84

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/jisa

Multi-tenant attribute-based access control for cloud infrastructure services

Canh Ngo *, *Yuri Demchenko*, *Cees de Laat*

Informatics Institute, University of Amsterdam, Science Park 904, 1098XH Amsterdam, The Netherlands

ARTICLE INFO

Article history:
Available online 23 December 2015

Keywords:
Access control
Multi-tenant
Attribute-based access control
Intercloud
Cloud computing

ABSTRACT

Cloud Computing is developed as a new wave of ICT technologies, offering a common approach to on-demand provisioning of computation, storage and network resources that are generally referred to as infrastructure services. Most of currently available commercial cloud services are built and organized reflecting simple relations between single provider and customers with the simple security and trust model. New architectural models should deliver multi-provider heterogeneous cloud services environments to organizational customers representing multiple user groups. These models need to be enforced by consistent security services operating in virtualized multi-provider cloud environment. They should incorporate complex access control mechanisms and trust relations among cloud actors. In this paper, we analyze cloud services provisioning use-cases and propose an access control model for multi-tenant cloud services using attribute-based access control model. We also extend the model for Intercloud scenarios with the exchanging tokens approach. To facilitate attribute-based policy evaluation and implementing the proposed model, we apply an efficient mechanism to transform complex logical expressions in policies to compact decision diagrams. Our prototype of the multi-tenant attribute-based access control system for Intercloud is developed, tested and integrated into the GEYSERS project. Evaluations prove that our approach has a good performance in terms of numbers of cloud resources and numbers of clients.

1. Introduction

Cloud Computing is effectively used to improve scalability, availability, elasticity and security of IT management in many application areas. It adopts advantages of many technologies such as virtualization, service-oriented architecture, and Utility computing to allow customers and providers to cut costs on system deployments and operations. Many studies and best practices documents related to clouds deployment, design, development, operations and management have been proposed to incorporate above technologies (Dillon et al., 2010; Foster et al., 2008; Fox et al., 2009; Hogan et al., 2011; Höfer and Karagiannis, 2011; Mell and Grance, 2011). Clouds in such approaches enable users' data to store on share virtualized cloud infrastructures, which are on-demand provisioned at providers' facilities. The virtualized infrastructures capacities can be elastically scaled up and down depending on varying users' demands. Cloud providers build up their systems based on the multi-tenant architecture (Chong et al., 2006; Garcia-Espin et al., 2012; Guo et al., 2007; Mell and Grance, 2011). Thus, security in general as well as access control for cloud service management should be aware of the multi-tenancy pattern in this architecture.

* *Corresponding author.* Informatics Institute, University of Amsterdam, Science Park 904, 1098XH Amsterdam, The Netherlands. Tel.: +31 20 525 6918.
E-mail address: t.c.ngo@uva.nl (C. Ngo).
http://dx.doi.org/10.1016/j.jisa.2015.11.005

In cloud resource management, resources in cloud are virtualized and managed in a common resource pool (Chong et al., 2006; Garcia-Espin et al., 2012; Ghijsen et al., 2013; Hogan et al., 2011; Mell and Grance, 2011). Depending on the stage in its life cycle, the resource may be administrated by one or multiple entities in the the multi-tenancy pattern:

- At the initial stage, resources are managed by the provider, who is the economic and management owner of available idle resources.
- When a tenant subscribes set of cloud resources, their economic and administrative ownerships will be transferred exclusively to this tenant during subscribed period.
- The subscribed tenant may want to allow accesses from its users, or due to collaboration requirements, share part of its resources to another trusted tenant with specific conditions like allowed actions, time, location.
- The trusted tenant in turn can manage the shared resources by defining access control policies for its users, or share to another one.

A use-case of multi-tenancy pattern could be as follows: a cloud provider offers cloud services to commercial companies (tenants) in which their employees can access subscribed cloud resources (e.g., storage, Virtual Machine (VM), databases, spreadsheets, etc.) during subscribing time. The provider should guarantee isolations between subcribed resources of tenants. On the other side, tenants may want to collaborate with each other by sharing resources. The commercial firm C, wants to audit its finance statements. It signs the contract with the auditing firm A to allow A's consultants can read-only to parts of C's resources during a specific time-frame.

According to Hogan et al. (2011), cloud services relies mainly on virtualization, multi-tenant architecture, elasticity and diversity of accesses. The characteristics of multi-tenancy pattern bring distinctions between Cloud Computing and previous distributed systems. For this reason, access control for clouds should be designed to support such scenarios.

The on-demand self-service and rapid elasticity properties in clouds (Mell and Grance, 2011) requires that the access control design must handle dynamic changes of entities as well as fine-grained authorization. For example, a typical cloud Infrastructure as a Service (IaaS) service provides different plans (e.g., storage size, speed, computing powers, bandwidth, lifetime, etc.) to customers in which the number of subscribers could reach thousands. Each of them can then manage hundreds of end-users. In these cases, numerous resource objects are provisioned over time with dynamic identifiers. Besides, clouds must handle accesses from users using diversity of clients in both types and numbers (e.g., mobile devices, laptops, workstations).

Traditional access control models are designed to manage accesses from subjects to objects with specific operations via authorization statements. A trivial statement is a triple of ⟨*subject, object, operation*⟩, in which the ⟨*object, operation*⟩ is known as a permission. Role-based Access Control (RBAC) approaches (ANSI, 2004; Ferraiolo et al., 2001; Sandhu et al., 1996) were introduced with roles as an abstraction layer decoupling subjects and permissions. RBAC was supported to apply in different areas, from stand-alone, enterprise-level or cross-enterprise applications. However, even the design purpose of RBAC is to large enterprise systems with even hundreds or thousands of roles and users in tens thousands (Sandhu et al., 1999), such systems may have problems on scalability in role and object explosions (Franqueira and Wieringa, 2012; Kuhn et al., 2010). Analysis in Franqueira and Wieringa (2012) estimates that RBAC should be used for systems with static structure where roles and hierarchy are clearly defined; entities individuality and locality are limited; and managed objects are stable. Large-scale cloud services management systems often have dynamics of provisioned pooling objects, varieties of entities and sophisticated fine-grained authorization regarding dynamical context-specific attributes, in which RBAC approaches may not be suitable.

To overcome limitations of RBAC systems, Attribute-based Access Control (ABAC) was identified with the central idea that access can be determined based on present attributes of objects, actions, subjects and environment in the authorization context (Hu et al., 2014; Jin et al., 2012; Yuan and Tong, 2005). The ABAC can be used to model RBAC as well as other traditional access control models (Jin et al., 2012) The fine-grained authorization feature of ABAC makes it more flexible and scalable than RBAC. Thus, ABAC is mostly suitable for cloud management services.

However, using large numbers of attributes in ABAC and the elasticity of clouds produce challenges in management and deployment. The complexity of attributes criteria in rules and conflict resolutions may arise during applying ABAC in access control for large-scale systems like cloud. ABAC implementation like eXtensible Access Control Markup Language (XACML) standard (OASIS, 2013) only limits at defining a general ABAC policy language but without indicating how to integrate with system resource information models for attribute management. There is also no related work on ABAC defining required constraints in policy composition and management for multiple authorities in multi-tenant systems.

With all such challenges and motivated by cloud and Intercloud scenarios analyses (GEANT, 2010; GEYSERS, 2010; Ngo et al., 2011, 2012), as well as related work on access control for clouds (Amazon, 2013; Bernal Bernabe et al., 2012; Bethencourt et al., 2007; Calero et al., 2010; Goyal et al., 2006; Jin et al., 2014; Sahai and Waters, 2005; Tang et al., 2013), we introduce the Multi-tenant Attribute-based Access Control (MT-ABAC) approach which formalize the ABAC applied for the multi-tenancy pattern. It not only aims to provide a scalable and flexible resources and entities management of the ABAC, but also contains related policy constraints facilitating delegations and collaborations among tenants and users in multiple levels. The extended model is applied for Intercloud scenarios with the exchanging tokens approach for fine-grained dynamic trust establishment. To facilitate attribute-based policy evaluation and implementing the proposed model, we apply an efficient mechanism to transform complex logical expressions in policies to compact decision diagrams. Our prototype of the multi-tenant access control system for Intercloud is developed, tested and integrated into the GEYSERS project. Evaluations demonstrate that our system has good performance in terms of number of cloud resources, clients and policies.