Future Generation Computer Systems 29 (2013) 497–508

Contents lists available at SciVerse ScienceDirect

Future Generation Computer Systems

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

Secure cooperative access control on grid

A. Merlo *

E-Campus University, Novedrate, Italy
Dipartimento interscuola di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi (DIBRIS), University of Genova, Italy

ARTICLE INFO

Article history:
Received 23 January 2012
Received in revised form
23 July 2012
Accepted 6 August 2012
Available online 14 August 2012

Keywords:
Grid computing
Cooperative access control
Grid security
Broadcast encryption

ABSTRACT

The access to Grid resources depends on policies defined by the administrators of the physical organizations and of the Grid middleware. This approach does not require support for access control in the middleware, but since changes in the access control policy of the Virtual Organization imply the involvement of one or more administrators, it lacks the flexibility needed in several Grid application scenarios. In this paper we propose a novel Cooperative Access Control model for Grid environments that increases the flexibility of the access control model offered by state-of-the-art Grid platforms without requiring changes in the middleware. The approach is based on collaboration among Grid users and allows them to exchange access permissions to Virtual Resources without the intervention of administrators. We also propose a solution based on Broadcast Encryption which allows to enforce a Cooperative Access Control model on Grids avoiding misuse and granting anonymity. Finally, we show that our solution can be defined on top of the access control mechanisms offered by state-of-the-art Grid middleware and illustrate how the proposed model has been implemented as a service in a service-oriented Grid environment.

1. Introduction

The Grid Computing paradigm aims at realizing a common distributed environment in which resources are shared and accessed by many users independently from the organizations the users and the resources belong to. Grid Computing has been motivated by the low level of resource utilization efficiency that generally afflicts the management of single administrative domains.

In fact, administrative domains (e.g. a company) usually make partial use of the available computational, storage and network resources and an effective usage of the available resources is commonly perceived as a key challenge [1].

Grid Computing middleware (e.g. Globus Toolkit 4 [2] – hereafter GT4 – and gLite [3]) tackles the problem by providing a virtualization layer that allows the creation and management of Virtual Organizations (VOs) on top of Physical Organizations (POs), i.e. the administrative domains. Physical Resources (PRs) (e.g. CPU, RAM, disk storage) in different POs are then mapped into VRs, thereby making them accessible outside the boundaries of the POs they belong to.

A user of a *PO* (e.g. a researcher in a University) needs a Grid User (*GU*) identity in order to access VRs. (A *GU* is uniquely identified by an identifier that constitutes an entry point to the Grid.) VRs are more complex structures than PRs and individual VRs usually comprise different PRs. For instance, an execution service (e.g. the GRAM service of the Globus Toolkit 4 [2]) is made of different PRs like disk storage, RAM and CPU. Similarly, a *GU* is associated with a set of Physical Users (PUs—e.g. accounts on different machines possibly belonging to different POs).

The mappings between GUs and PUs and between VRs and PRs are kept in specific files and structures in current Grid middleware (e.g. the `grid-mapfile` in the Globus Toolkit) that can be manipulated only by the local Grid administrator (e.g. the *globus user*, a non-privileged account that has access to configuration files and structures).

By means of virtualization, Grid middleware extends the visibility and accessibility of PRs; however, this unavoidably complicates access control.

In a *PO* the access control policy is managed by an administrator who directly specifies the access privileges that registered users have on the available PRs.

In a VO access control is inherently related to the definition of mappings between PRs and VRs as well as GUs and PUs. Moreover the PRs associated with the VRs are subject to the security policies defined by the administrator of the respective POs.

Thus, the resulting access control policy at the Grid layer depends on the access control policy at the Physical layer as well as on the mappings that relate GUs and VRs with PUs and PRs respectively, as exemplified in Fig. 1.

* Correspondence to: DIBRIS, University of Genova, Genova, Italy. Tel.: +39 0103532344.
E-mail addresses: alessio.merlo@uniecampus.it, alessio.merlo@unige.it.

0167-739X/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2012.08.001

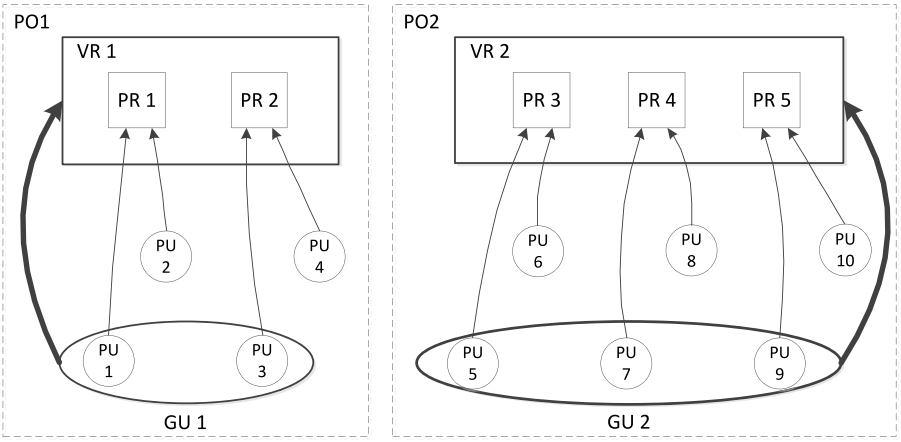

Fig. 1. Narrowness of the cross-layer Grid access control. GU_1 is granted access to VR_1 since her physical identities (PU_1 and PU_2) are granted access to the physical resources composing VR_1 (i.e. PR_1 and PR_2). Nevertheless, GU_1 is not allowed to access VR_2 due to the lack of access privileges on PR_3, PR_4 and PR_5.

This approach does not require support for access control in the middleware, but since changes in the access control policy of the VO imply the involvement of the administrators of the POs participating in the VO and/or of the local Grid administrator, it lacks the flexibility needed in several application scenarios. For instance, in collaborative environments (e.g. research institutions) it is desirable to give GUs the freedom to share their access to VRs with other trusted GUs at their discretion. Also, in mission-critical applications (e.g. [4,5]) unexpected events may require access to VRs for a *GU* that would not be normally allowed and that security policies as defined at the POs and/or Middleware layers would prevent.

In this paper we propose a Cooperative Access Control (CAC) model for Grid environments increasing the flexibility of the access control model offered by state-of-the-art Grid platforms without requiring changes in the middleware. Our approach is based on collaboration among Grid users at a granularity of single permission and it allows users to share access permissions to VRs without the intervention of administrators.

Besides, we show that our model can be defined on top of the access control mechanisms offered by state-of-the-art Grid middleware. This makes our approach non invasive and adaptable to different middlewares. In fact, non invasiveness frees the middleware from the burden of implementing proper connectors or extensions; as a consequence, the CAC model can be adapted to different general purpose middlewares.

Furthermore, we provide a security assessment of the CAC model, highlighting security issues related to its enforcement on actual Grids. Then, we propose a solution based on Broadcast Encryption (i.e. BE-CAC) which allows to utilize CAC in a secure way, avoiding misuses and granting anonymity. Finally, we describe an actual implementation of BE-CAC on a GT4-based Grid.

Structure of the paper. In Section 2 we introduce the access control model in Grid environments and discuss its limitations. In Section 3 we present the Cooperative Access Control model (CAC) as an extension to current Grid access control. In Section 4 we assess security issues related to the enforcement of CAC model, and we provide a solution based on Broadcast Encryption (BE-CAC) which allows to securely implement CAC model on actual Grids. To this aim, in Section 5 we describe our prototype implementation of BE-CAC on top of the Globus Toolkit 4 middleware in a non invasive way. Finally, in Section 6 we discuss the related works and in Section 7 we draw some conclusions and future works.

2. Modeling Grid access control

Three layers are involved in the authorization decisions in Grid environments: the physical layer, the middleware layer and the Grid layer.

- *Physical layer.* The level where users and resources are part of a single administrative domain. At this level, PUs are granted access to PRs according to the access control policy of the POs and enforcement of the access control policies is thus left to POs. For instance, a cluster of computers running the UNIX operating system will rely on the UNIX access control mechanisms to implement and enforce the access control policy of the *PO* they belong to. Let $PO_1, \ldots, PO_n$ be POs that participate in the VO. We model the access control policy of PO_i as the triple $PAC_i = \langle PR_i, PU_i, PA_i \rangle$, where PR_i is the set of PRs, PU_i is the set of PUs, and $PA_i \subseteq (PR_i \times PU_i)$ is the *permission assignment relation* of PO_i, for $i = 1, \ldots, n$. We assume that the sets of PUs and PRs in different domains are mutually disjoint, i.e. that $PR_i \cap PR_j = \emptyset$ e $PU_i \cap PU_j = \emptyset$ for all $i, j = 1, \ldots, n$ with $i \neq j$. We define $PAC = \langle PR, PU, PA \rangle$, where $PR = \bigcup_{i=1}^{n} PR_i$, $PU = \bigcup_{i=1}^{n} PU_i$ and $PA = \bigcup_{i=1}^{n} PA_i$.
- *Middleware layer.* The Middleware layer is responsible for virtualizing the PRs of the single administrative domains into VRs of the VO. The middleware layer keeps track of the user correspondence between GUs and PUs as well as the resource correspondence between VRs and PRs:
  - *User mapping:* $UR = \langle GU, PU, UM \rangle$, where *GU* is the set of GUs in the VO, $PU = \bigcup_{i=1}^{n} PU_i$ is the set of PUs in the different POs and $UM \subseteq (GU \times PU)$ is the *user mapping relation*. In GT4 such mapping is stored in the `grid-mapfile` and is only modifiable by the *globus* user.
  - *Resource mapping:* $RR = \langle VR, PR, RM \rangle$, *VR* is the set of VRs in the VO, $PR = \bigcup_{i=1}^{n} PR_i$ is the set of PRs and $RM \subseteq (VR \times PR)$ is the *resource mapping relation*. In GT4, this mapping is defined in internal structures of the middleware and in the Monitoring and Discovering Service (MDS). The MDS is a basic service of GT4 that acts as an index of the services publicly available in the Grid.
- *Grid layer.* The Grid layer consists of the grid users (GUs) and the virtual resources (VRs) of the VO. At this layer, access control amounts to deciding whether any given $gu \in GU$ is entitled to access a $vr \in VR$. This depends on whether the PUs associated with *gu* have enough permissions to get access to the PRs associated with *vr* according to the access control policies of the respective POs. We model the access control policy at this layer by the triple $GAC = \langle GU, VR, GA \rangle$, where *GU* and *VR* are the sets of GUs and VRs of the VO respectively and $GA \subseteq (GU \times VR)$ is the *permission assignment relation* at the Grid level and is such that $(gu, vr) \in GA$ if and only if for all *pr* such that $(vr, pr) \in RM$ there exists *pu* such that $(gu, pu) \in UM$ and $(pu, pr) \in PA$. If $(gu, vr) \in GA$, then we say that *gu is granted access to vr according to GAC*. The access control policy at the Grid layer