Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science
ELSEVIER
Electronic Notes in Theoretical Computer Science 186 (2007) 3–26
www.elsevier.com/locate/entcs

High Level Conflict Management Strategies in Advanced Access Control Models

Frédéric Cuppens[a] Nora Cuppens-Boulahia[a]
Meriam Ben Ghorbel[a,b]

[a] *GET/ENST Bretagne, 2 rue de la Châtaigneraie, 35512 Cesson Sévigné Cedex, France*

[b] *SUPCOM Tunis, Route de Raoued Km 3.5, 2083 Ariana, Tunisie*

Abstract

Specifying a security policy that includes both permissions and prohibitions, may lead to conflicts. This corresponds to a situation where a subject is both permitted and prohibited to perform a given action on a given object. We adopt a comparative approach to investigate this problem. We first investigate access control models based on rules, called Rule-BAC, and present weaknesses that arise when we try to manage conflicts in this model. In particular, Rule-BAC models fail to provide decidable solution to redundant rules and potential conflicts problems. Then, we show how a more structured model, say OR-BAC (Organization Based Access Control), gifted with inheritance mechanism make redundant rules and potential conflict problems tractable in polynomial time.

Keywords: Rule-BAC, Or-BAC, Potential Conflict, Prioritized policy.

1 Introduction

Access control is modelled as a set of authorizations specified either by a security officer or a private user in accordance with some security policy. Usually, these authorizations specify that a given subject (a user or a process) is permitted to perform a given action (an access mode) on a given object (a resource of the system). This static authorization triple $\langle subject, action, object \rangle$ is suited for traditional environments and applications but is less appropriate to meet requirements of the rising systems. Indeed, there is a need of more expressiveness, that is other kinds of authorizations must be supported: content based authorization, constraint based authorization and more particularly negative authorization. In that way the security officer is given means to specify general contextual permission rules and associate exceptions to these general rules using prohibitions. For instance, a nurse may be permitted to consult a medical record (general rule) except the physician's private comments (exception corresponding to prohibition). Moreover, in an access control

doi:10.1016/j.entcs.2007.01.064

model where hierarchies and inheritance mechanisms are included, prohibitions can be used to regulate the inheritance policy of permissions.

However, when an access control model includes the possibility to specify both permissions and prohibitions, some conflicts may occur. This is the case when a subject is both permitted and prohibited to perform an action on an object. Hence, the system might not be able to decide either to allow or deny the access. This problem was investigated by several models (for instance [2,9,7,3,16,5,15,1]). The conflict issue must be addressed by defining a conflict resolution strategy. A conflict resolution strategy consists in a set of rules that enable the system to decide, in case of a conflict, to discard either the positive or the negative authorization. Therefore, the resulting access control policy will depend on the chosen conflict resolution policy. Thus the security officer should have the possibility to define his or her own conflict resolution strategy in order to obtain a relevant access control policy.

Rule based access control (Rule-BAC) termed models [7,3,15,4,13] take the lead of access control models attempting to meet expressiveness requirements and offer means to solve conflict problems. In this kind of model, access control is defined as a set of rules $Condition \rightarrow Authorization$ where $Condition$ is a set of constraints over the subjects, actions and objects. In this paper, we begin with analyzing conflict management in the context of Rule-BAC Model and show that there are several problems this model fails to solve. First, assigning higher priorities to some access rules to manage conflicts can lead to the emergence of rules that never apply, say redundant rules. Unfortunately, checking the non-redundancy condition is undecidable and there is a lack of replacement solution in the Rule-BAC model. The second unsolved problem is that current solutions only manage *actual* conflicts but they are unable to detect *potential* conflicts, that is the coexistence of rules that lead to some conflicts if their associated conditions are simultaneously satisfied. Managing potential conflicts is important since we gain the guarantee that actual conflict will never occur. However, checking the potential conflict condition in the Rule-BAC model is also undecidable. We argue that the main reason of these drawbacks is a structure lack of Rule-BAC models.

We then analyze these problems in the context of the Or-BAC model [17] and show how all of them are formally and effectively solved. In Or-BAC, which is a structured and more expressive model built on top of Rule-BAC, specifying an access control policy is centered around the concept of *organization*. Each organization can specify its own security policy at an "organizational" level. For this purpose, a policy specifies that some *roles* are permitted or prohibited to perform some *activities* on some *views*. These concepts of role, activity and view are used to specify the policy independently from concrete implementation of subjects, actions and objects in the system. Moreover, Or-BAC model gives means to specify contextual authorizations.

The approach used to manage conflicts in Or-BAC is based on assigning priorities to access control rules as we suggest in Rule-BAC. Nevertheless, to overcome difficulties encountered in Rule-BAC, we restate the problems of rule redundancy and potential conflicts using inheritance mechanisms and separated constraints specification [14]. We then show that, using this approach, rule redundancy and potential