A Practical Framework for Policy Composition and Conflict Resolution

Ousmane Amadou Dia, Computer Science and Engineering Department, University of South Carolina, Columbia, SC, USA

Csilla Farkas, Computer Science and Engineering Department, University of South Carolina, Columbia, SC, USA

ABSTRACT

In collaborative environments where resources must be shared across multiple sites, the access control policies of the participants must be combined in order to define a coherent policy. The relevant challenge in composing access policies is to deal with inconsistencies or modality conflicts. This difficulty exacerbates when the policies to compose are specified independently by different entities with no global power to decide in case of conflicts which entity must take precedence. This paper presents a semi-automated framework called Policy Composition and Conflict Resolution framework (P2CR) to address this issue. They focus on access control policies expressed as XACML statements. The authors propose a three-level conflicts resolution strategy: i) by using metadata added to the policies, ii) by using a defeasible logic theory, and iii) by providing recommendations to the entities owners of the resources. First, they provide a mechanism to add metadata to XACML. Second, they combine the access policies without prioritizing any of the entities involved in the composition. Given the context of the authors’ work, they consider this approach to be more suitable than the current approaches that are mainly negotiation-oriented or assign priorities to the policies. Finally, the resulting composite policy appears flexible and easily adjustable to runtime conflicts.

Keywords: Annotations, Conflict Detection, Conflict Resolution, Defeasible Logic, OWL, Policy Composition, XACML

1. INTRODUCTION

Recent years have witnessed a growing number of special-purpose communities in which different organizations (or tenants) with common interests and needs interact and share pools of configurable resources governed by a cloud service provider. Community clouds have many advantages. They enable organizations that are technologically different and geographically separated to collaborate in a seamless manner. However, they can be difficult to manage especially when the tenants have differing access policies. The diversity of the policies of the entities may lead to serious obstacles in establishing a safe collaboration within the cloud. An important requirement for precisely achieving this goal is that each entity, tenant

DOI: 10.4018/jsse.2012100101
as well as cloud service provider, abides by
the security, compliance and risk management
requirements of the others. Thus, to allow the
entities to interact safely, their access policies
must necessarily be compared and composed.

In this paper, leveraging the community
clouds as an illustrative example, we address
the policy composition problem in a broader
scenario in which different entities are interested
in composing their independently stated policies
while retaining their autonomy i.e., maintaining
the control over their resources. A non-trivial
challenge generally faced in this context is the
occurrence of conflicts. Two access policies may
apply to same objects and yield upon request
of the objects contradictory evaluation results.
Access control systems governed by such poli-
cies cannot deterministically decide whether to
grant access to the requested objects or to deny
the access. Consequently, they may even allow
certain users to access resources they are not
authorized for or deny the access to the legiti-
mate ones. Thus, to enable access policies in
individual systems to unambiguously evaluate
users requests, many conflict resolution strate-
gies have been proposed (Reeder, Bauer, Cranor,
Reiter, & Vaniea, 2009; Cuppens, Cuppens-
Boulahia, & Ghorbel, 2007; Dong, Russello,
& Dulay, 2008; Jajodia, Samarati, Sapino, &
Subramanian, 2001; Moffett & Sloman, 1993;
XACML, 2005).

However, in situations where several
autonomous entities want to integrate their
independent access policies, these strategies
are limited. Conflicts that occur in this scenario
are difficult to eliminate because of the diver-
sity of the policies of the entities, and more
importantly because of the conflict resolution
strategies that they use. Currently, no effective
technique exists for resolving these conflicts
while the policies are being integrated (Mohan
& Blough, 2010). An intuitive approach could
however be to pick the conflict resolution
strategy of a random entity and adopt it as the
conflict resolution technique of all the policies.
Unfortunately, because each entity enforces
the strategy it finds more suitable to its needs,
such an approach would result in many cases
inconclusive. A typical example is two entities,
A that applies the Deny-overrides (XACML,
2005) scheme to restrict access to its resources,
and B that uses the Permit-overrides (XACML,
2005) method to ensure the availability of its
data. In this case, if the strategy that B uses is
applied, then resources of A may be accessed
by unauthorized users. Conversely, if we opt
for the strategy of A, then access to resources
of B may be severely restricted.

Over the past years, considerable work in
composing independently stated access policies
has been done (Bertolissi & Fernandez, 2008;
Bonatti, Vimercati, & Samarati, 2000; Bruns,
Dantas, & Huth, 2007; Lin, Rao, Bertino, &
Lobo, 2010; Lupu & Sloman, 1999; Mazzoleni,
Bertino, & Crispo, 2008; Ni, Bertino, & Lobo,
2009; Rao, Lin, Bertino, Lui, & Lobo, 2009).
The approach common to many of the proposed
studies is to combine these policies based on
the priorities they are assigned with. Assigning
priorities to policies is however difficult, and
understanding them even more because the
priorities are generally represented as numbers
and no semantic is attached to them to reflect
their meaning (Agrawal, Giles, Lee, & Lobo,
2007; Lee, Boyer, Olson, & Gunter, 2006). In
addition, in many studies, the composite policy
resulting from the integration of the policies is
enforced in only one point. What this entails is
either the party that administers the single point
of enforcement is heading all the entities, or it is
mandated by them to combine their individual
access policies and to manage the resulting
policy. However, entities that are interested
in combining their policies may be under the
authority of different parties or reluctant to
part with the administration of their resources.
Lastly, in many proposals, conflicts are detected
manually (e.g., Agrawal et al., 2007) and their
causes usually overlooked (e.g., Mazzoleni et
al., 2008), and in order to eliminate the conflicts,
access to resources to which conflicting policies
apply are denied. However, such an approach is
limited. First, without a precise knowledge of
what causes a conflict it is difficult to guarantee
the effectiveness of the solution that one would
adopt to resolve the conflict. Moreover, in a
Related Content

Architecture-Centered Integrated Verification
www.irma-international.org/chapter/architecture-centered-integrated-verification/51970/

Improving Security and Safety Modelling with Failure Sequence Diagrams
www.irma-international.org/article/improving-security-safety-modelling-failure/64193/

Model-Driven Development of Mobile Information Systems
www.irma-international.org/chapter/model-driven-development-mobile-information/77708/

Fault Injection for On-Board ERTMS/ETCS Safety Assessment
www.irma-international.org/chapter/fault-injection-board-ertms-etcss/66670/

ART-Improving Execution Time for Flash Applications
Ming Ying and James Miller (2011). International Journal of Systems and Service-Oriented Engineering (pp. 1-20).
www.irma-international.org/article/art-improving-execution-time-flash/55059/