Chapter 7
An Integrated Approach for the Enforcement of Contextual Permissions and Pre-Obligations

Yehia Elrakaiby
TELECOM Bretagne, France

Frédéric Cuppens
TELECOM Bretagne, France

Nora Cuppens-Boulahia
TELECOM Bretagne, France

ABSTRACT

Pre-obligations denote actions that may be required before access is granted. The successful fulfillment of pre-obligations leads to the authorization of the requested access. Pre-obligations enable a more flexible enforcement of authorization policies. This paper formalizes interactions between the obligation and authorization policy states when pre-obligations are supported and investigates their use in a practical scenario. The main advantage of the presented approach is that it gives pre-obligations both declarative semantics using predicate logic and operational semantics using Event-Condition-Action (ECA) rules. Furthermore, the presented framework enables policy designers to easily choose to evaluate any pre-obligation either (1) statically (an access request is denied if the pre-obligation has not been fulfilled); or (2) dynamically (users are given the possibility to fulfill the pre-obligation after the access request and before access is authorized).

DOI: 10.4018/978-1-4666-2163-3.ch007
INTRODUCTION

Traditional security policy systems provided a simple yes/no answer to access requests. However, it was recognized that access often depends on some user-actions being performed before access is granted. For instance, an access rule may specify that users are allowed to download music files provided that they pay $1 first. In this case, if a user requests to download, for example, the latest single of Muse, s/he is asked to pay $1. If the payment is made successfully, the user is allowed to download the requested file. Such requirements are called pre-obligations. Neither traditional access control models such as DAC (NCSC, 1987) and RBAC (Ferraiolo & Kuhn, 1992) nor more recent contextual security models such as ASL (Jajodia, Samarati, & Subrahmanian, 1997) and OrBAC (Abou El Kalam et al., 2003) support pre-obligations: In these models, an access request is only allowed if the conditions associated with a permission authorizing the access are true when the access request is made.

There are several advantages of supporting pre-obligations in the policy language. First, this provides additional expressiveness since it enables policy administrators to specify that subjects may fulfill some of the access requirements after the access request. Furthermore, it separates the expression of requirements from the functional specification (the code) of the application. Thus, the analysis of policy requirements is simplified and administrators are able to modify the behavior of the system by updating policy rules without recoding the application.

To support pre-obligations, a number of works (Bettini, Jajodia, Wang, & Wijesekera, 2002, 2003; Ni, Bertino, & Lobo, 2008) subordinate obligations to access control rules. This approach has some limitations. For instance, obligations are only activated after access requests and general obligations are not supported. In addition, this approach generally produces intricate access control policies since permissions and obligations are often specified within the same rule. This is the approach used in (Ni et al., 2008) to specify permissions and their associated pre-obligations. The main limitation of previous works on pre-obligations is however that none formalized the effects of supporting pre-obligations on the evolution of the authorization and obligation policy states. This is essential to provide a deeper understanding of pre-obligations and their enforcement in information systems. In addition, this formal approach allows the study and the analysis of change in the authorization and obligation policy states in the presence of pre-obligations. Therefore, it enables, for instance, to derive plans to reach some particular authorization states (Becker & Nanz, 2008; Craven et al., 2009) or to explain the deactivation of pre-obligations after permission activation.

In this paper, we study the specification and the enforcement of pre-obligations. In our approach, we formalize the enforcement of pre-obligations using an extension of the language “Lactive” (Baral & Lobo, 1996). Lactive enables the description of change in state using concepts from action specification languages (Gelfond & Lifschitz, 1993). Thus, it enables reasoning about state evolution and the study of interactions between pre-obligations and the authorization and obligation policy states. Lactive also supports the specification of reactive behavior using active rules. This feature enables us to provide formal operational semantics for the enforcement of pre-obligations.

To simplify the expression of pre-obligations in access control rules, we specify pre-obligations in the form of contexts. A security rule context (Cuppens & Cuppens-Boulahia, 2008) denotes a set of conditions which have to be true for the security rule to be effective. For instance, a context during_working_hours may hold (be true) every working day from 8 in the morning until 6 in the afternoon. In our approach, context rules may be used to specify requirements which state that some user-action should be taken. These contexts are called pre-obligation contexts. We support