Secure Software Engineering Based on Business Process Modeling

Joseph Barjis, Delft University of Technology, The Netherlands

ABSTRACT

Security requirements must be tackled early in software design and embedded in corresponding business process models. As a blueprint for software design, business process models complemented with security requirements will prevent many security breaches. To accomplish secure business process modeling, the underlying method must adhere to certain capabilities and capture actions, actor roles, and interactions. The resultant models should lend themselves to automatic analysis (simulation) to ensure captured security requirements are correctly aligned with the process flow. Thus, the tradeoff between the level of security and business performance can be studied before actual software design. Since unauthorized actions cause security breaches, the software the system's social setting could be a cradle for defining security requirements. Security requirements can be identified based on the roles, authorities, and obligations of the social actors using the system. This paper introduces a method for security embedded business process modeling. The proposed method draws on two well-tested theoretical foundations—enterprise ontology and organizational semiotics.

Keywords: *DEMO Methodology, Information System Security, Norm Analysis Method, Organizational Semiotics, Petri Net, Secure Business Process Modeling (Secure BPM), Security Driven Business Process Modeling*

1 INTRODUCTION

Business process modeling plays a pivotal role in software system design (Barjis, 2008), which is often referred to as a software system blueprint (Nagaratnam et al., 2005). Due to this important role, business process models are becoming a natural focus for incorporation of security requirements that consequently should be passed to the software design and development phases. Business process modeling allows software designers to capture functional requirements better and more easily while automatically generating these requirements from the business process models (Basin et al., 2003). Especially with the prevailing process-centric approach in software design, business process models are the appropriate departure point for understanding the business domain for which a software system is developed, and for understanding the social setting in which the software system will be used.

Failing to recognize this importance up front may cause many security requirements to be overlooked and left for end-of-pipe solutions.

DOI: 10.4018/jsse.2010040101

The lack of appropriate security requirements, such as authorized access control in the activities carried out by the actors, leaves the software system as well as the whole enterprise vulnerable to possible threats (D'Aubeterre et al., 2008). In regard to the social environment of a software system, identification and definition of behavioral rules serve as a valuable source of security requirements. Despite this obvious importance, software system design is mainly driven by functional requirements.

Systems requirements define the functional aspects of a system such as what a system is supposed to achieve, therefore often these functional requirements dominate the business requirements and constraints even if the latter are well defined (Khan et al., 2004). The fact that the business requirements and constraints are not captured in the corresponding conceptual models in any form for later usage, and most of them are not available for further consultation at the implementation stage, leaves the chance of omitting essential security constraints wide open. In order to avoid this pitfall, Khan et al. (2004) suggest (diagrammatically) incorporating security requirements into the model and making them part of the created artifact (model). In this manner, the security requirements identified at the business level will better make their way to the later phases of software systems design.

In this paper, we introduce a business process modeling method that captures security functions during the business process modeling phase. Actually, this approach has been discussed and tested in many past studies, where existing methods have been extended with the capability of capturing security requirements, for example Baskerville (1988), Herrmann and Pernul (1999), Backes et al. (2003), Mana et al. (2003). A similar approach is also proposed in D'Aubeterre et al. (2008), where security requirements are incorporated into the business process model, or in the works that resulted in enriched Use Case (Siponen et al., 2006), where the UML Use Case diagram is extended to incorporate security requirements in the design phase. Actually, UML, as the *de facto* industry standard, has been widely favored by the proponents of secure business process modeling researchers (Lodderstedt et al., 2002; Firesmith, 2003; Jurjens, 2004; Basin et al., 2006).

Despite diligent efforts made by these researchers, security-driven business process modeling is still far from becoming a true departure point for secure software development and therefore it is not an adapted practice in companies (Neubauer et al., 2006). One of the dominant reasons the role of business process modeling in software engineering is not realized is that the existing methods suffice to address merely conceptual and semantic levels and, therefore, they present little pragmatic value for software designers. By using the existing methods, it is difficult to automatically analyze the models and, therefore, the embedded security functions are not possible to test and simulate.

The core contribution of this paper is introduction and discussion of a modeling method developed for secure business process modeling (secure BPM) that incorporates not only the related business transactions but also security functions with the resultant model being based on formal semantics. The starting point for the proposed modeling method is identification of business transactions, then the roles and interactions of the actors as they carry out the assigned operations, tasks, and actions.

In comparison to the existing methods for incorporation of security requirements in business process models, the method proposed in this paper offers certain advantages:

1. The proposed method results in a security-embedded business process model that is completely based on formal semantics. That is, the models can be automatically analyzed or simulated to study the impact of the incorporated security requirements and whether the security requirements compromise business performance in any way.
2. The proposed modeling method is based on the theoretical foundation laid out in the studies of enterprise ontology (Dietz, 2006), which is proven to achieve