Towards Generating Richer Code by Binding Security Abstractions to BPMN Task Types

Julio Cesar Damasceno[1], Bruno Leonardo Barros Silva[1], Bruno Leonardo Barros Silva[1], Robson Wagner Albuquerque de Medeiros[1], Fernando Antonio Aires Lins[1], Nelson Souto Rosa[1], Paulo Romero Martins Maciel[1]

Bryan Stephenson[1], Hamid Reza Motahari Nezhad[2], Jun Li[2], Caio Northfleet[2]

Abstract: This paper presents an approach for binding security requirements to different BPMN task types to create secure executable business processes.

1 Introduction

Annotating conceptual models enables generating richer (sometimes executable) code from models. There is an increasing interest in incorporating security requirements into business process models encoded in BPMN (Business Process Modeling Notation) specifications. This paper presents an extension that allows the binding of security requirements to different BPMN task types.

2 Binding Security Abstractions to BPMN Task Types

A BPMN Task is an atomic activity included within a business process definition. BPMN includes eight different types of tasks: Service, Receive, Send, User, Script, Manual, Reference and None. Security requirements may be expressed and bound to these task types by using the following abstractions [1]: NF-Attribute that models non-functional characteristics such as confidentiality; NF-Action that models design decisions, algorithms, data structures, and configurations which implement security enforcement mechanisms to achieve an NF-Attribute; NF-Statement that models constraints defined on an NF-Attribute to guide decisions taken to implement an NF-Attribute. For example, NF-Statement "High Confidentiality" may require the implementation of the NF-Action UseCryptography

[1] HP Labs Palo Alto
{bryan.stephenson,hamid.motahari,jun.li@hp.com}
[2] HP Brazil
{caio.northfleet@hp.com}

choosing public key-based encryption algorithm. In this work, we define how the security abstractions may be bound to various Task types. The following table illustrates task types and how security abstractions are bound to them. The implementation of these bindings is in progress and partially implemented in a modelling tool demonstrated in [1].

Type	Description	Binding
Service	It represents a service (web service or application). A task receives a message, performs processing and sends a message to mark the completion of the task.	NF-Actions may be assigned to input and output messages and processing, e.g., the NF-Action *UseCryptography* bound to *Service* Task means that input/output messages must be encrypted prior to entering or leaving the task.
Receive	This task waits for a message to arrive from an external participant (relative to the Business Process).	NF-Actions associated to *Receive* tasks are applied to the input message, e.g., the NF-Action *UseCryptography* bound to a Receive task means that the input messages must be encrypted by the external participant.
Send	This task sends a message to an external participant (relative to the Business Process).	NF-Actions apply to the sent message, e.g., the NF-Action *UseCryptography* bound to a *Send* task means that the input sent message must be encrypted before being sent.
User	Task performed by a human with the assistance of a software application.	Bindings applied to *Service* tasks are applicable to this task.
Script	Task executed by a business process engine by interpreting a script.	NF-Actions must be implemented by the engine, e.g., the NF-Action *UseCryptography* bound to a *Script* task means that the engine must encrypt any data manipulated by the script.
Manual	Task performed without the aid of any business process execution engine or any application.	NF-Action may not be bound to this type of task as they are not computationally implemented.
Reference	Refers to another task with similar behavior.	NF-Actions binding is defined according to the referenced Task type.

3 Conclusion

In this paper, we show the application of using abstractions for security requirements to annotate elements of the BPMN conceptual model. We show that this enables generating richer code (e.g. WS-BPEL code) when service composition is modeled using BPMN.

Acknowledgement. This research is supported by Hewlett-Packard Brasil Ltda. using incentives of Brazilian Informatics Law (Law nº 8.248 of 1991).

4 References

[1] A. Souza et al. “Incorporating Security Requirements into Service Composition: From Modeling to Execution”. In 7th International Joint Conference on Service Oriented Computing (ICSOC 2009). Stockholm, Sweden.