From Early Requirements Analysis towards Secure Workflows *

Ganna Frankova[1], Fabio Massacci[1], and Magali Seguran[2]

[1] DIT - University of Trento
email: {ganna.frankova,fabio.massacci}@unitn.it
[2] SAP Labs France, SAP Research - Security and Trust
email: magali.seguran@sap.com

Abstract. Requirements engineering is a key step in the software development process that has little counterpart in the design of secure business processes and secure workflows for web services. This paper presents a methodology that allows a business process designer to derive the skeleton of the concrete coarse grained secure business process, that can be further refined into workflows, from the early requirements analysis.

1 Introduction

There are many requirements engineering frameworks for modeling and analysing security requirements, such as *SI**/Secure Tropos, UMLsec, MisuseCase, and AntiGoals. There are several methodologies aim to web services and business processes design [8, 4, 9]. We noticed that there is a gap among the requirements engineering methodologies and the actual production of software and business processes based on a Service-Oriented Architecture (SOA). Business processes and security issues are developed separately and often do not follow the same strategy [6]. The existing design methodologies for web services do not address the issue of developing secure business processes and secure workflows. An overview of approaches aimed to use requirements engineering methodologies in the context of web services can be found in [1]. There are a number of security standards in the area of SOA. For instance, WS-Federation defines the mechanisms for federating trust, WS-Trust enables security token interoperability, WS-Security covers the low level details such as message content integrity and confidentiality. The question we address in this paper is "How to obtain a secure workflow from the early requirements analysis?".

We address the issue of secure workflows design based on early requirements analysis by presenting a methodology that bridges the gap between early requirements analysis and secure workflows for web services development. We introduce a language for secure business processes description, which is a dialect of WS-BPEL for the functional parts and abstracts away low level implementation details from WS-Trust, WS-Security and WS-Federation specifications.

* This work has been partly supported by the IST-FP6-IP-SERENITY project

2 The *SI**/Secure Tropos framework

*SI**/Secure Tropos is a formal framework and a methodology for modelling and analysing security requirements [2, 5]. In this work we employ the early security requirements analysis to design secure business process and secure workflow. *SI**/Secure Tropos uses the concepts of actor and goal. Actors can be *agents* or *roles*. *SI**/Secure Tropos also supports the notion of *delegation of permission* and *delegation of execution* to model the transfer of entitlements and responsibilities from an actor to another. *Trust of permission* and *trust of execution* are used to model the expectation of one actor about the behavior and capabilities of another actor. The meaning of trust of permission is that a trustor trusts that trustee will at least fulfill a service while trust of execution means that trustor trusts that trustee will at most fulfill a service, but will not overstep it.

From a methodological perspective, *SI**/Secure Tropos is based on the idea of building a model of the system that is incrementally refined and extended. Specifically, goal analysis consists of refining goals and eliciting new social relationships among actors. They are conducted from the perspective of single actors using AND/OR decomposition. In case an actor does not have the capabilities to achieve his own objectives or assigned responsibilities by himself, he has to delegate them to other actors making their achievement outside his direct control.

3 Secure workflows design based on early requirements

A secure business process is originated by the early requirements analysis and then is used to the development of an appropriate workflow.

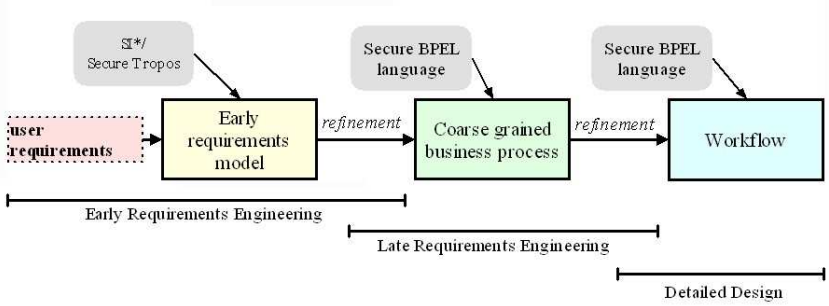

Fig. 1. Relations among early requirements, business process and workflow levels.

The process of deriving a secure workflow from early requirements is presented in Figure 1. The process includes three phases, namely, (1) early requirements engineering, (2) late requirements engineering and (3) detailed design.

Early requirements engineering. During early requirements analysis the domain actors and their dependencies on other actors for goals to be fulfilled are identified. For early requirements elicitation, one need to reason about trust relationships and delegation of authority. We employ *SI**/Secure Tropos modelling framework to derive and analyse both functional dependencies and security and trust requirements. Various activities contribute to the acquisition of

the early requirements model, namely:
Actor modelling aims at identifying actors and analysing their goals.
Functional dependency modelling aims at identifying actors depending on other actors for obtaining services, and actors which are able to provide services.
Permission delegation modelling aims at identifying actors delegating to other actors the permission on services.
Trust modelling aims at identifying actors trusting other actors for services, and actors which own services.

A graphical representation of the model obtained in these modelling activities is given through the actor, functional dependency, authorization, and trust diagrams [2], respectively.

Late requirements engineering. Late requirements engineering is concerned with a definition of the functional and non-functional requirements of the system-to-be. In this work the proposed refinement methodology aims to obtain an appropriate coarse grained business process and workflow based on early requirements. The refinement is processed by diagrams created in the early requirements engineering phase. The methodology takes the components of the diagrams and derives a secure business process constructs from them that is described by the proposed Secure BPEL language.

Considering actor diagram, the notion of actor is refined into partner in Secure BPEL, a root goal is refined into business process while AND/OR goal decomposition with delegation are refined into orchestration. We assume that each actor has a single root goal that can be decomposed by AND/OR goal decomposition. The notions of delegation of execution and delegation of permission presented in dependency and authorization diagrams are refined into choreography of services and authorization respectively. As for trust diagram, trust on execution and permission are refined into choreography of attestation that is further refined into attestation of integrity for the notion of trust on execution and attestation of reporting for trust on permission. The concept of attestation characterizes the process of vouching for the accuracy of information [3]. Attestation of integrity provides proof that an actor can be trusted to report integrity and performed using the set or subset of the credentials associated with the actor. Attestation of reporting is the process of attesting to the contents of integrity reporting.

Secure BPEL Language. Secure BPEL is a language for secure business processes and workflows description. Secure BPEL is a dialect of Web Services Business Process Execution Language (WS-BPEL) [7] for the functional parts and abstracts away low level implementation details from WS-Security and WS-Federation specifications.

For the lack of space we do not present the details of Secure BPEL in this paper, refer to [1] for the language description and illustration with a typical loan origination process scenario. Here we do not have space to present the whole scenario. We focus on the concept of delegation of execution that is relevant to the security requirement such as separation of duties and introduce the refinement of the dependency diagram in order to give an example. The loan

origination process describes a customer applying for a loan to the BBB bank. Several external ratings conducted by the Credit Bureau need to be obtained by the processing clerk in order to check the credit worthiness of the customer. Delegation of execution appears when the processing clerk delegates the external rating analysing to the Credit Bureau. The concept of delegation of execution is refined as follows. At the delegater side, the partner processing clerk invokes the service creditWorthinessCheck (by the `<invoke>` construct) from the partner Credit Bureau. While at the delegatee side, the partner Credit Bureau, the delegatee responds to a service invocation (the `<pick>` construct) accepting the message of service invocation and execute the creditWorthinessCheck service.

4 Concluding remarks

The main contribution of the paper is to bridge the gap between early requirements analysis and the design of secure workflows based on SOA. In particular, we have proposed a methodology that allows to derive the concrete secure business processes from the early requirements analysis. Furthermore, the secure business processes are refined in order to obtain the appropriate secure workflows that can be described by the proposed language for secure business processes description called Secure BPEL. The proposal is illustrated with an e-business banking case study, a working scenario of the SERENITY project.

The research presented in this work is still in progress. Currently we are diving into the details of the low level secure requirements of messages integrity and confidentiality that will be included in the next release of the language.

References

1. G. Frankova, F. Massacci, and M. Seguran. From Early Requirements Analysis towards Secure Workflows. Technical report, University of Trento, 2007.
2. P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone. Requirements Engineering for Trust Management: Model, Methodology, and Reasoning. *International Journal of Information Security*, 5(4):257–274, October 2006.
3. Trusted Computing Group. TCG Specification Architecture Overview Revision 1.2, April 2003.
4. D. Lau and J. Mylopoulos. Designing Web Services with Tropos. In *Proceedings of IEEE International Conference on Web Services*, San Diego, USA, July 6-9 2004.
5. F. Massacci, J. Mylopoulos, and N. Zannone. An Ontology for Secure Socio-Technical Systems. *Handbook of Ontologies for Business Interaction*, 2007.
6. T. Neubauer, M. Klemen, and S. Biffl. Secure Business Process Management: A Roadmap. In *Proceedings of International Conference on Availability, Reliability and Security*, Vienna, Austria, April 2006.
7. OASIS. Web Services Business Process Execution Language Version 2.0, August 2006. Public Review Draft, `http://docs.oasis-open.org/wsbpel/2.0/`.
8. M.P. Papazoglou and J. Yang. Design Methodology for Web Services and Business Processes. In *Proceedings of the International Workshop on Technologies for E-Services*, Hong Kong, China, August 2002.
9. L. Penserini, A. Perini, A. Susi, and J. Mylopoulos. From Stakeholder Needs to Service Requirements. In *Proceeding of International Workshop on Service-Oriented Computing: Consequences for Engineering Requirements*, Minneapolis, Minnesota, USA, September 2006.