SeCMER: A Tool to Gain Control of Security Requirements Evolution*

Gábor Bergmann[1], Fabio Massacci[2], Federica Paci[2], Thein Tun[3], Dániel Varró[1], and Yijun Yu[3]

[1] DMIS - Budapest University of Technology and Economics
{bergmann,varro}@mit.bme.hu
[2] DISI - University of Trento
{fabio.massacci,federica.paci}@unitn.it
[3] DC - The Open University
{t.t.tun,y.yu}@open.ac.uk

Abstract. This paper presents SeCMER, a tool for requirements evolution management developed in the context of the SecureChange project. The tool supports automatic detection of requirement changes and violation of security properties using change-driven transformations. The tool also supports argumentation analysis to check security properties are preserved by evolution and to identify new security properties that should be taken into account.

Keywords: security requirements engineering, secure i*, security argumentation, change impact analysis, security patterns.

1 Introduction

Requirements change continuously making the traceability of requirements hard and the monitoring of requirements unreliable. Moreover, changing requirements might have an impact on the satisfaction of security properties a system design should satisfy.

In this paper we present SeCMER[1], a tool for requirement evolution management which provides three main features: a) *Modelling requirement evolution*: The drawing of requirement models in different state of the art requirement languages such as SI* [5], Problem Frames (PF) [7] and SeCMER[2] is supported; b) *Argumentation-based security analysis* [7]: it allows the requirement engineer to check that security properties are preserved by evolution and to identify new security properties; c) *Change management based on evolution rules* [3]: it allows to detect changes into the requirement model, to check argument validity, to automatically detect violations or fulfilment of security properties, and to issue alerts prompting human intervention.

* Work parly supported by the project EU-FP7-ICT-FET-IP-SecureChange.

[1] A detailed description of the tool implementation is reported in [4], and a demonstrating screencast is presented at http://www.youtube.com/watch?v=OWwzcNeSuJM

[2] SeCMER is a requirement language that includes concepts belonging to SI*, PF and security such as asset.

W. Abramowicz et al. (Eds.): ServiceWave 2011, LNCS 6994, pp. 321–322, 2011.

These capabilities of the tool are provided by means of the integration of Si* [5] as a graphical modeling framework for security requirements, OpenPF [6] which supports argumentation analysis, and EMF-IncQuery [2] which supports change detection.

2 Demo Scenario

We illustrate the features supported by our prototype using as example the ongoing evolution of ATM systems planned by the ATM 2000+ Strategic Agenda [1] and the SESAR Initiative. We focus on the introduction of the Arrival Manager (AMAN), which is an aircraft arrival sequencing tool to support air traffic controllers, and an IP based data transport network called System Wide Information Management (SWIM) that should replace actual point-to-point networks.

1. **Requirements evolution.** We show how the tool allows to model the evolution of the requirement model as effect of the introduction of the SWIM.
2. **Change detection based on evolution rules.**
   a *Detection of a security property violation based on security patterns.* We show how the tool detects that the integrity security property of the resource "Meteo Data" is violated due to the lack of a trusted path.
   b *Automatically providing corrective actions based on evolution rules.* We show how evolution rules may suggest corrective actions for the detected violation of the integrity security property.
3. **Argumentation-based security analysis.** We show how argumentation analysis is used to provide evidence that the information access property applied to the meteo data is preserved by evolution.

References

1. EUROCONTROL ATM Strategy for the Years 2000+ Executive Summary (2003)
2. Bergmann, G., Horváth, Á., Ráth, I., Varró, D., Balogh, A., Balogh, Z., Ökrös, A.: Incremental evaluation of model queries over EMF models. In: Petriu, D.C., Rouquette, N., Haugen, Ø. (eds.) MODELS 2010. LNCS, vol. 6394, pp. 76–90. Springer, Heidelberg (2010)
3. Bergmann, G., et al.: Change-Driven Model Transformations. Change (in) the Rule to Rule the Change. Software and System Modeling (to appear, 2011)
4. Bergmann, et al.: D3.4 Proof of Concept Case Tool, `http://www.securechange.eu/sites/default/files/deliverables/D3.4%20Proof-of-Concept%20CASE%20Tool%20for%20early%20requirements.pdf`
5. Massacci, F., Mylopoulos, J., Zannone, N.: Computer-aided support for secure tropos. Automated Software Engg. 14, 341–364 (2007)
6. Tun, T.T., Yu, Y., Laney, R., Nuseibeh, B.: Early identification of problem interactions: A tool-supported approach. In: Glinz, M., Heymans, P. (eds.) REFSQ 2009 Amsterdam. LNCS, vol. 5512, pp. 74–88. Springer, Heidelberg (2009)
7. Tun, T.T., et al.: Model-based argument analysis for evolving security requirements. In: Proceedings of the 2010 Fourth International Conference on Secure Software Integration and Reliability Improvement (2010)