Security Requirements Engineering Framework for Software Product Lines

Daniel Mellado[1], Eduardo Fernández-Medina[2] and Mario Piattini[2]
[1] Spanish Tax Agency,
Madrid (Spain)
damefe@esdebian.org
[2] University of Castilla La-Mancha, Institute of Information Technologies & Systems, Dep. of Information Technologies & Systems
Paseo de la Universidad, 4 13071 Ciudad Real (Spain)
(Eduardo.FdezMedina or Mario.Piattini)@uclm.es

Abstract. Context: The correct analysis and understanding of security requirements are important because they assist in the discovery of any security or requirement defects or mistakes during the early stages of development. Security requirements engineering is therefore both a central task and a critical success factor in product line development owing to the complexity and extensive nature of software product lines (SPL). However, most of the current SPL practices in requirements engineering do not adequately address security requirements engineering. **Objective**: The aim of this approach is to describe a holistic security requirements engineering framework with which to facilitate the development of secure SPLs and their derived products. It will conform with the most relevant security standards with regard to the management of security requirements, such as ISO/IEC 27001 and ISO/IEC 15408. **Results**: This framework is composed of: a security requirements engineering process for SPL (SREPPLine) driven by security standards; a security reference meta model to manage the variability of those SPL artefacts related to security requirements; and a tool (SREPPLineTool) which implements the meta model and supports the process. **Method**: A complete explanation of the framework will be provided. The process will be formally specified with SPEM 2.0 and the repository will be formally specified with an XML grammar. The application of SREPPLine and SREPPLineTool will be illustrated through a description of a simple example as a preliminary validation. **Conclusion**: Although there have been several attempts to fill the gap between requirements engineering and SPL requirements engineering, no systematic approach with which to define security quality requirements and to manage their variability and their related security artefacts in SPL models is, as yet, available. The contribution of this work is that of providing a systematic approach for the management of the security requirements and their variability from the early stages of product line development in order to facilitate the conformance of SPL products with the most relevant security standards.

Keywords: Product lines; Common Criteria; ISO/IEC 27001; Security requirement; Security requirements engineering.

1. Overview of SREPPLine

SREPPLine (Security Requirements Engineering Process for Software Product Lines) [1] is an iterative and incremental process which is an add-in of tasks that can be incorporated into and tailored to an organization's SPL development process model to provide it with a security requirements engineering approach. We have defined the key tasks that must be part of each SPL activity, signifying that the order in which the steps are performed depends on the particular process that is established in an organisation. The activities and their tasks can thus be combined with existing development methods. It can therefore be termed as a scalable process since not all the tasks and steps are required, and developers could create their own lightweight process by selecting a subset of the steps in each task.

It is a security-feature or security-goal based process which is driven by risk and security standards (specifically ISO/IEC 27001 and Common Criteria (CC)). It deals with security requirements and their related artefacts from the early phases of SPL development in a systematic and intuitive manner especially tailored to SPL based development. It is based on the use of the latest and most widely validated security requirements techniques, such as security use cases or misuse cases, along with the CC components and ISO/IEC 27001 controls into the SPL lifecycle in order to facilitate SPL product security certification (depicted as cylinders in the centre of Fig. 1). Our proposed process suggests the use of a method to carry out risk assessment which conforms to

ISO/IEC 13335. It specifically uses Magerit, for both SPL risk assessment and SPL products risk assessment. The aim of SREPPLine is to minimize both knowledge of the necessary security standards and security expert participation during SPL product development. To this end, it provides a Security Reference Model (shown in Fig. 1) to facilitate security artefact reuse and to implement the Security Reference Meta Model. This meta model is composed of the Security Variability Sub-Meta Model and the Security Requirement Decision Sub-Meta Model, both of which assist in the management of the variability and traceability of the security requirements related artefacts of the SPL and its products. The meta model is the basis used by the SREPPLine tasks to capture, represent and share knowledge about security requirements for SPL and help to certify them against security standards. In essence, it is a knowledge repository with a structure to support security requirements reasoning in SPL.

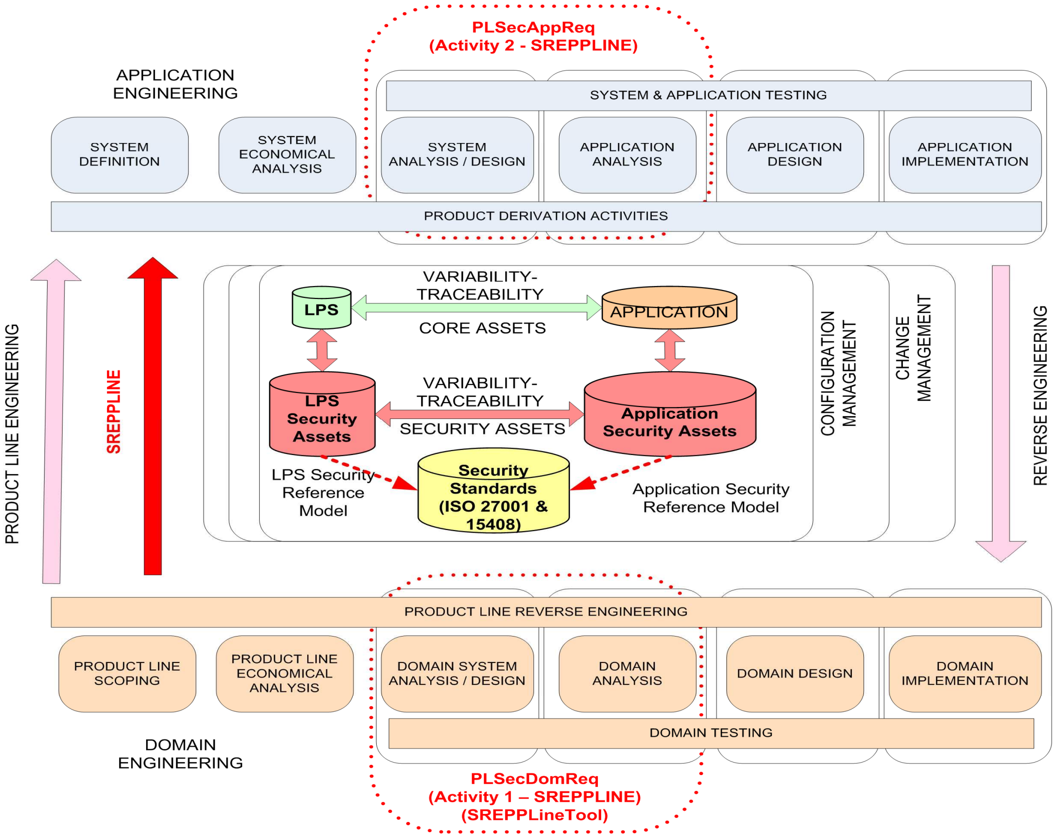

Fig. 1 Security requirements engineering framework for Software Product Lines (SPLs or LPS)

Fig. 1 shows the typical activities of both application engineering (at the top of the figure) and domain engineering (at the bottom of the figure) based on the framework for SPL engineering proposed by Bühne et al. (shown in rounded squares in Fig. 1).

Finally, the management of these repositories is performed by the prototype tool that we have developed to provide automated support to SREPPLine, SREPPLineTool. This prototype implements the Security Reference Meta Model by means of dynamic repositories of security artefacts, and guides us in the execution of the process in a sequential manner. This tool thus permits us to apply the SREPPLine process in an SPL development by providing automated support to its activities

1. D. Mellado, E. Fernández-Medina, and M. Piattini, *Security requirements engineering framework for software product lines.* Information and Software Technology, 2010. **52**: p. 1094-1117.