Volume 30, Issue 6, August 2008

ISSN: 0920-5489

ELSEVIER

COMPUTER STANDARDS & INTERFACES

SPECIAL ISSUE

State of Standards in the Information Systems Security Area

Guest Editors: Eduardo Fernández-Medina and Mariemma I. Yagüe

The International Journal on the Development and Application of Standards for Computers, Software Quality, Data Communications, E-topics, Interfaces and Measurement

COMPUTER STANDARDS & INTERFACES (CS&I)

The International Journal on the Development and Application of Standards for Computers, Software Quality, Data Communications, Interfaces and Measurement

---

Aims and scope

The quality of software, well-defined interfaces (hardware and software), the process of digitalisation, and accepted standards in these fields are essential for building and exploiting complex computing, communication, multimedia and measuring systems. Standards can simplify the design and construction of individual hardware and software components and help to ensure satisfactory interworking.

COMPUTER STANDARDS & INTERFACES is an international journal dealing specifically with the topics below.

The journal:

- provides information about activities and progress on the definition of computer standards, software quality, interfaces and methods, at national, European and international levels
- publishes critical comments on standards and standards activities
- disseminates user's experiences and case studies in the application and exploitation of established or emerging standards, interfaces and methods
- offers a forum for discussion on actual projects, standards, interfaces and methods
- stimulates relevant research by providing a specialized refereed medium.

COMPUTER STANDARDS & INTERFACES is concerned with the specification, development and application of standards and with high-level publications of developments and methods in the following areas:

- ***Standards, Information Management, Formal Methods*** - Computers, Processors, Storage, Operating systems, Languages, Databases, Graphics, User interface, Multimedia, Information security, Office automation, Development of standards and instruments, Applications
- ***Software Quality, Software Process*** - Languages, Operating systems, Programming, Requirements specification, Design & implementation, Inspection & test, Maintenance, Product and process evaluation, Performance, Tools, Metrics, Embedded systems, Software in measurement and technical systems including real-time aspects, Development of International Standards in Software Engineering
- ***Distributed Systems, Open Systems, E-Topics*** - Digital interfaces, System and device buses, Fieldbuses, Data communication, Distributed computing, Protocols, Open systems interconnection, Local and wide area networks, Internet, Worldwide Web, Network security, Cryptology, E-services, E-business, E-commerce
- ***Data Acquisition*** - Analog-to-digital conversion, Specification, Modelling, Industrial electronics, Real-time systems, Laboratory automation, Automatic measurement, Process control, Electromagnetic compatibility
- ***Digital Instruments Standardization*** - Forum of EUPAS, European Project for ADC-based devices Standardisation (IMEKO TC-4 Working Group on A/D and D/A Converter Metrology), IEEE TC-10, IEC TC-42/WG8, IEC TC-85/WG16; Standardisation of specifications, modelling, testing, and analog and digital processing for digital instruments

The last issue of a volume includes an Author index and a Subject index.

CS&I also covers general topics concerning the standardization process, such as technical, political and commercial aspects of standards, their impact on the marketplace, cost/benefit analyses, legislative issues, and relationships among national and international standards bodies.

COMPUTER STANDARDS & INTERFACES
Volume 30/6

ELSEVIER

Amsterdam • Boston • Jena • London • New York • Oxford • Paris •
Philadelphia • San Diego • St. Louis

Printed in Printforce, Alphen a/d Rijn, The Netherlands

♾ The paper used in this publication meets the requirements of ANSI/NISO 239.48-1992 (Permanence of Paper)

COMPUTER STANDARDS & INTERFACES

Special Issue:

State of standards in the information systems security area

Guest Editors:

Eduardo Fernández-Medina
Mariemma I. Yagüe

Available online at www.sciencedirect.com
ScienceDirect
Computer Standards & Interfaces 30 (2008) 339–340
COMPUTER STANDARDS & INTERFACES
www.elsevier.com/locate/csi

Guest Editorial

State of standards in the information systems security area

The development and use of standards in information technologies, and in particular, in the area of security, have grown up in the last years. The main reason is the increasing need for interoperability due to the new scenarios (e.g. collaborative work, heterogeneous IT processes and systems) that have emerged on the Web.

As standards represent an important means of achieving interoperability on the WWW and the Web has become a new global platform, the scientific community focuses its attention on the different international standards bodies and organizations, such as the National Institute of Standards and Technology (NIST), the International Standard Organization (ISO), the International Electrotechnical Commission (IEC), the Institute of Electrical and Electronics Engineers (IEEE), the International Telecommunication Union (standardization section — ITU-T), the Organization for the Advancement of Structured Information Standards (OASIS), the Internet Engineering Task Force (IETF), the World Wide Web Consortium (W3C), etc. Not only specifications from these organisms become standards but also recommendations from very representative consortiums, such as the Object Management Group (OMG) become de facto standards as well.

These standards and specifications are crucial in many areas related to security in information systems. First of all, standards are important for unifying security techniques in different aspects, such as security protocols, cryptography, access control, authentication, privacy, integrity, attack detection, availability, personal data protection, etc. Secondly, an organizational view of security is very relevant too. Therefore, standards also reach this approach through the definition of security management systems, security maturity models, risk management guidelines, and so on. Nevertheless, standards are intensively considered in software engineering processes for developing information systems, and these standards are constantly redefined and extended in order to incorporate security into the software development. In this sense, standards and specifications define software modeling techniques, requirement engineering techniques, architecture languages and specifications, pattern definitions, life cycles models, software development methodologies, metamodeling techniques, etc., and also many specific technologies, with specific security requirements, such as Web services, grid computing, mobile devices, etc.

This Special Issue of the International Journal of Computer Standards and Interfaces includes a selection of the most representative papers presented at the Fifth International Workshop on Security in Information Systems (WOSIS 2007), which was held in Funchal, Madeira — Portugal, June 12–13, 2007. This edition of the workshop has been specially oriented to standards for security in information systems, obtaining a representative sample of the existing papers dealing with security and where standards fulfill a relevant role.

Our workshop has matured year by year, and it is established as a forum for high quality research papers in the area of security in information systems. The most valuable assets of this workshop to be attractive for authors are a very exclusive set of program committee members, along with the invitation of exceptional speakers, highly relevant in this scientific area. Among them, we can mention, for example, Yvo Desmedt, Sushil Jajodia, Ernesto Damiani, Leonardo Chiariglione, and Ruth Breu. Additionally, selections of the best papers of past editions of the workshop have been published in international journals such as Information Systems Security, Journal of Research and Practice in Information Technology, and Internet Research.

In the following paragraphs, a brief introduction to each selected paper will be stated.

The first contribution by Soler et al., presents an extension to the CWM (Common Warehouse Metamodel) specification developed by the OMG (Object Management Group) with the aim of specifying security in data warehouse models at the PSM (Platform Independent Model) level of the Model Driven Architecture. In this paper, standards such as UML, CWM, MOF, QVT, MDA are intensively used in the area of software engineering, with the purpose of integrating security into the development of data warehouses.

The second contribution by Tafreschi et al., deals with a reputation system, which, on the one hand, facilitates trust building among business partners who interact in an ad hoc manner with each other, and on the other hand enables market participants to rate the business performance of their partners as well as the quality of the offered goods. In this proposal, many types of standards and specifications, such as HTTP, XML, SOAP and WSDL are directly and indirectly used for the definition of the system architecture.

The third paper by Mellado et al., states a security standard-based process for software product line development. The proposal is a contribution in the area of security requirements engineering for software product lines, but providing its integration

0920-5489/$ - see front matter
doi:10.1016/j.csi.2008.03.001

with the Common Criteria (ISO/IEC 15408), as well as with some of the most relevant standards related to security management, such as ISO/IEC 17799 and ISO/IEC 27001. This proposal also conforms to IEEE 830-1998, regarding software requirements specification.

The fourth manuscript by Agreiter et al., puts forward a framework that provides fair non-repudiation for Web services messages, since there is not any sophisticated standard specifying this requirement for this environment. However, the paper deals with several standards, specifications, and protocols, such as UML, XML, SOAP, SSL, WSS, TTP, XACML, etc.

The fifth contribution, by Damiani et al., specifies a general query rewriting technique to securely query XML, the standard for data interchange. The proposed model is described by a Deterministic Finite Automata and is able to rewrite unsafe queries into safe ones, avoiding the many backtrackings inherent to non-deterministic finite automata. The proposed technique is linear with the size and depth of the repository schema.

The sixth contribution, by Plößl and Federrath, deals with security requirements of vehicular ad hoc networks (VANET). Nodes (mainly vehicles) are expected to communicate by means of the North American DSRC standard that makes use of the IEEE 802.11p standard for wireless communication. Authors evaluate some requirements such as message integrity and non-repudiation as well as propose a security infrastructure meeting all requirements, specially designed to protect privacy of the VANET users and efficient in terms of computational needs and bandwidth overhead.

The seventh contribution, by Canfora and Visaggio, refers to privacy preservation in highly dynamic, untrustworthy and scalable contexts, implementing the paradigm of front end trust filter. Therefore, the proposed solution makes the assumption that a privacy policy can be expressed at least at three different levels of detail, so-called layers, in other words, the statement of the policy, the strategies for realizing such policy and the implementation, which applies the strategy at the level of applications and database. This three-layered structure confers a high degree of flexibility.

The eighth contribution, by Zych, et al., studies the key management problem of the data-centric protection model, where data is cryptographically protected and allowed to be outsourced or even freely float on the network. Namely, when data is encrypted, the access control policies have to be taken into account so that control regulating what users are allowed to access to what data is maintained. Authors propose an efficient method that eliminates, as compared to broadcast encryption methods, the need for multiple copies of data keys and reduces to a single key the storage required per user. The solution is based on the Diffie–Hellman Key Exchange protocol, standardized by the RSA Laboratories as the Diffie–Hellman Key Agreement Standard.

The ninth contribution, by Sánchez, et al., is focused on the authorization problem. It shows how the eduroam user federation for an inter-NREN network roaming service based on AAA servers and the IEEE 802.1X standard can take advantage of the use of authorization services with the objective of offering a more grained network access control process. For that purpose, this work presents how eduroam can be extended with the NAS-SAML infrastructure and eduGAIN. The first is a network access control approach based on the AAA architecture and authorization attributes and the SAML and XACML standards. Secondly, the main goal of eduGAIN is to build an interoperable authentication and authorization infrastructure to interconnect different existing federations.

The tenth paper by Prandini and Ramilli proposes a communication scheme for remote system administration aimed at overcoming some intrinsic security issues of the traditional client–server models. While the subject of system administration has not been the subject of a comprehensive standardization activity, this proposal provides a viable alternative to de facto standards in the area of remote access such as SSH (RFC4250–4254), IPSec (RFC4301–4303 and related ones). Furthermore, it is related to the general problem of authentication and access control as defined in ISO/IEC 10181-2/3. The proposed system is based on human-oriented meeting places such as IRC (RFC2810–2813), but future extensions can foresee the design of more structured distributed meeting places, for instance, those in accordance with the CORBA Security Service definition.

This Special Issue does not try to cover all applications of security standards in information systems, since it would be impossible. However, we hope to offer a good sample of papers to show how important the use and development of standards for information technologies, and particularly to security are.

We would like to gratefully acknowledge the hard work and kindness of all members of our international program committee when performing their timely, complete and professional reviews. We would like to thank Sabrina De Capitani di Vimercati (Italy), Ernesto Damiani (Italy), Csilla Farkas (USA), Eduardo B. Fernández (USA), Steven Furnell (UK), Christian Geuer-Pollmann (Germany), Paolo Giorgini (Italy), Ehud Gudes (Israel), Carlos Gutiérrez (Spain), Haralambos Mouratidis (England), Jan Jürjens (Germany), Stamatis Karnouskos (Germany), Antonio Maña (Spain), Martin Olivier (South Africa), Brajendra Panda (USA), Günther Pernul (Gerrmany), Mario Piattini (Spain), Joachim Posegga (Germany), Indrajit Ray (USA), Indrakshi Ray (USA), Damian Sauveron (France), Ambrosio Toval (Spain), Rodolfo Villarroel (Chile), and Duminda Wijesekera (USA).

Finally, we would like to thank Computer Standards & Interfaces and Elsevier, and particularly Professor Bhavani Thuraisingham for giving us the opportunity to publish this Special Issue.

Eduardo Fernández-Medina
ALARCOS Research Group,
Information Systems and Technologies Department,
University of Castilla-La Mancha, Paseo de la Universidad 4,
13071 Ciudad Real, Spain
Corresponding author.
E-mail address: Eduardo.FdezMedina@uclm.es.

Mariemma I. Yagüe
GISUM Research Group, Department of Computer Science
Málaga University Campus Universitario de Teatinos,
s/n 29071 Málaga, Spain
E-mail address: mariemma@lcc.uma.es.

Computer Standards & Interfaces 30 (2008) 361–371

Contents lists available at ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi

Towards security requirements management for software product lines: A security domain requirements engineering process

Daniel Mellado [a,*], Eduardo Fernández-Medina [b], Mario Piattini [b]

[a] Ministry of Work and Social Affairs; Social Security IT Department, Software Development Centre of the National Social Security Institute; Madrid, Spain
[b] ALARCOS Research Group, Information Systems and Technologies Department, University of Castilla-La Mancha, Paseo de la Universidad 4, 13071 Ciudad Real, Spain

ARTICLE INFO

Available online 8 March 2008

Keywords:
Product lines
Common Criteria
ISO/IEC 27001
ISO/IEC 17799
Security requirement
Security requirements engineering
ISMS

ABSTRACT

Security and requirements engineering are one of the most important factors of success in the development of a software product line due to the complexity and extensive nature of them, given that a weakness in security can cause problems throughout the products of a product line. The main contribution of this work is that of providing a security standard-based process for software product line development, which is an add-in of activities in the domain engineering. This process deals with security requirements from the early stages of the product line lifecycle in a systematic and intuitive way especially adapted for product line based development. It is based on the use of the latest security requirements techniques, together with the integration of the Common Criteria (ISO/IEC 15408) and the ISO/IEC 17799 controls into the product line lifecycle. Additionally, it deals with security artefacts variability and traceability, providing us with a Security Core Assets Repository. Moreover, it facilitates the conformance to the most relevant security standards with regard to the management of security requirements, such as ISO/IEC 27001 and ISO/IEC 17799. Finally, we will illustrate our proposed process by describing part of a real case study, as a preliminary validation of it.

1. Introduction

Our society has become increasingly IT-based [34], depending as it does on a huge number of software systems which have a critical role and which manage critical and sensitive information, it is absolutely vital that Information Systems (IS) are properly assured from the very beginning [1,24], due to the potential losses faced by organizations that put their trust in all these IS. Moreover, it is widely-accepted the principle which establishes that the building of security at the early stages of the development process is cost-effective and also brings about more robust designs [18].

Furthermore, nowadays, there is an increase in the demand as well as in the complexity of the software needed. Thus, in order to obtain high-quality IS along with higher productivity, software product line (SPL) based development has become the most successful approach in the reuse field, because it can help us significantly reduce time-to-market as well as development costs [3,4], by increasing the reuse of all types of artefacts, thanks to the combination of coarse-grained components with a top-down systematic approach where software components are integrated into a high-level structure.

Due to the complexity and extensive nature of product line development, security and requirements engineering are much more important for product line practice. Security is a cross-cutting concern in software intensive systems and should consequently be subject to careful requirements analysis and decision making.

In addition the requirements for cost-effective product line development complicate this task. Therefore, the discipline known as Security Requirements Engineering is a very important part of the SPL development process for the achievement of secure SPL and applications/products, because it provides techniques, methods and standards for tackling this task in the development lifecycle. It also implies the use of repeatable and systematic procedures to ensure that the set of requirements obtained is complete, consistent, easy to understand and analysable by the different actors involved in the development of the system [19].

In the last few years, it has been a spectacular growing of security standards and security related proposals which have been developed to try to help develop security critical IS. Moreover, SPL reference architectures for security and SPL requirements management approaches and tools, such as [15,32] have recently been developed. Nevertheless, after analysing the previously performed comparative analyses of several relevant proposals of IS security requirements, as those of [6,23,25,29,31,33,35], etc. in [27,28], we conclude that those standards and proposals are neither specific enough for a systematic and intuitive treatment of SPL security requirements, nor make it easy the task of integrating security requirements engineering activities

* Corresponding author.
E-mail addresses: Daniel.Mellado@uclm.es (D. Mellado), Eduardo.FdezMedina@uclm.es (E. Fernández-Medina), Mario.Piattini@uclm.es (M. Piattini).

0920-5489/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.csi.2008.03.004

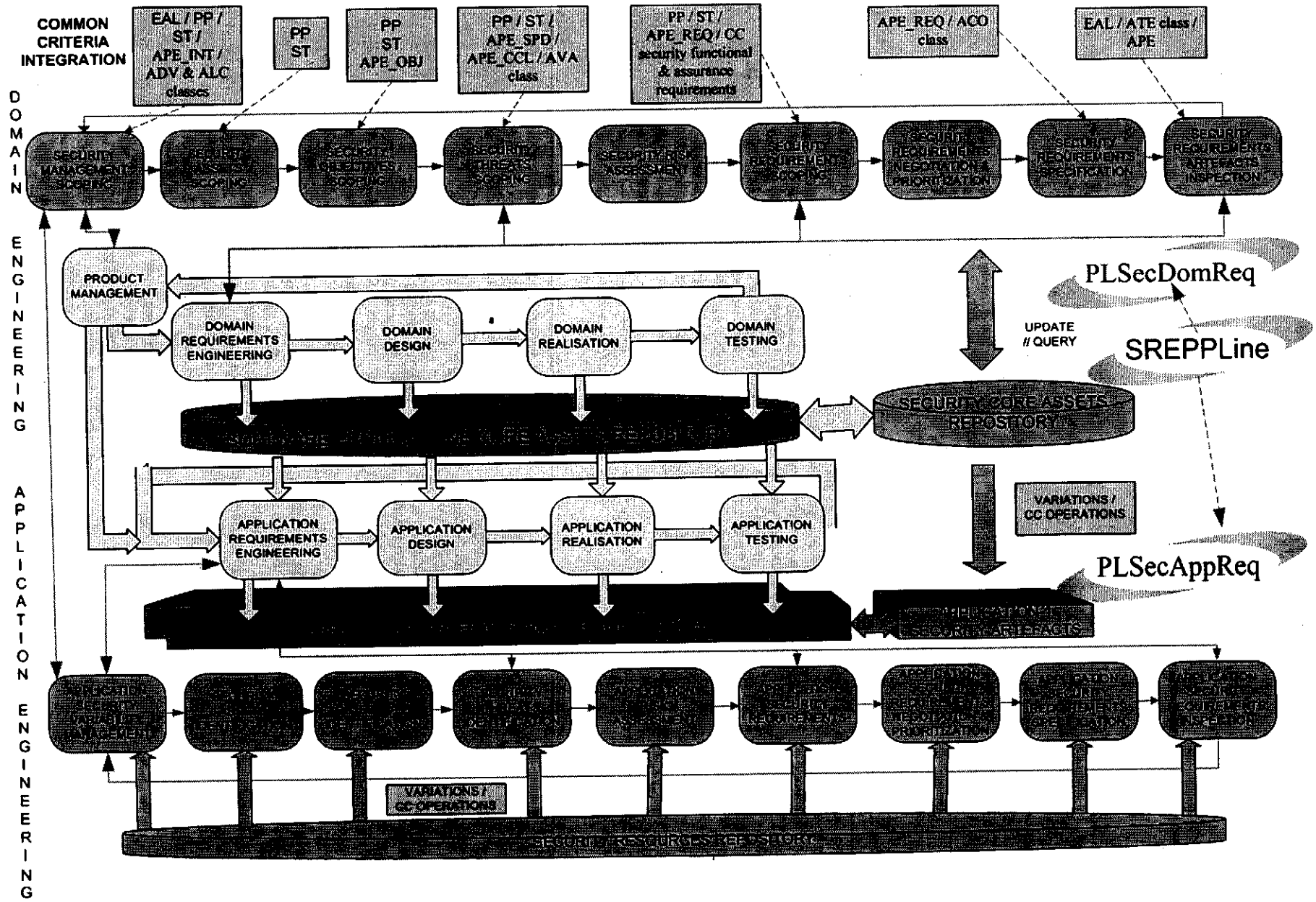

Fig. 1. Software product line security requirements engineering framework.

into the SPL based development. In addition, they do not provide intuitive, systematic and methodological support for the management of security requirements, with the aim of developing secure SPL and products that conform to the most relevant security standards with respect to the management of security requirements (such as mainly ISO/IEC 15408 [11] as well as ISO/IEC 27001 [13], ISO/IEC 17799 [12] or ISO/IEC 21827 [8]).

In this paper, as an evolution of our previous proposal SREP [28], we will present a Security Requirements Engineering Process for software Product Lines (SREPPLine), which is a standard-based process that describes how to integrate security requirements into the software engineering process in a systematic and intuitive way, as well as a simple integration with the rest of requirements and the different phases/processes of the SPL development lifecycle. Additionally, this process will facilitate the fulfilment of the IEEE 830:1998 standard [7], and it will help develop IS which conform to the aforementioned security standards with regard to the management of security requirements, and without being necessary to perfectly know those standards; hence, reducing the participation of security experts to achieve it. In order to reach these goals, our approach is based on the reuse of security artefacts which are integrated into the variability model of the SPL, by providing a Security Core Assets Repository (SCAR), together with the integration of the Common Criteria (CC) (ISO/IEC 15408) into the SPL lifecycle.

The rest of the paper is organized as follows: in Section 2, we will summarize the main concepts of requirements engineering in SPL. Then, in Section 3, we will present our proposed security requirements engineering process for SPL (SREPPLine). Next, in Section 4, we will explain the security core assets repository. Later, in Section 5 we will provide a small example of SREPPLine application in a real scenario. Finally, in Section 6, we will state our conclusions and sketch our future work.

2. A summary of product line requirements engineering

A software product line is a set of software-intensive systems sharing a common, managed set of features[1] [16] that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way [4].

The software product line engineering paradigm differentiates two processes: domain engineering and application engineering [30]. Domain engineering is the process of SPL engineering in which commonality and variability of the product line are defined and realised, and it is composed of five key sub-processes: product management, domain requirements engineering, domain design, domain realisation and domain testing. According to [30],the domain requirements engineering sub-process encompasses all activities for eliciting and documenting the common and variable requirements of the product line. Hence, domain requirements engineering differs from requirements engineering for single systems in the following subjects: commonality analysis; variability analysis and modelling, and the fact that it tries to anticipate prospective changes in requirements, such as laws, standards, technology changes and market needs for future applications. Application engineering is the process of SPL engineering in which the applications of the product line are built by reusing domain artefacts and exploiting the product line variability, and it is composed of the following sub-processes: application requirements engineering, application design, application realisation and application testing.

Product line requirements define the products and common and variable features of them in the product line. Requirements common across the entire family, which constitute the product line requirements and an important core asset, should be managed separately from requirements that are particular to a subset of the products (or to a single product), which must be managed as well. Product line requirements engineering includes different requirements sources, such as stakeholders (domain experts, etc.), existing products or competitors' products, existing model domains or the SPL scope. The SPL scope binds the products included in the product line: product line requirements refine the scope by more precisely defining the characteristics of the products in the product line. The scope and SPL requirements are tightly coupled and evolve together [4].

Finally, after analysing several relevant proposals of SPL requirements and some SPL requirement management tools, as those of [4,15,17,30,32], we argue that those proposals are neither specific enough for a systematic and intuitive treatment of SPL security requirements, nor make it easy the task of integrating security requirements engineering activities into the SPL based development.

3. SREPPLine: A security requirements engineering process for software product lines

The Security Requirements Engineering Process for software Product Lines (SREPPLine) is an add-in of activities (that are decomposed into tasks which receive input artefacts and which generate output artefacts, with the participation of different roles) that are integrated into a SPL development process model of any organization providing it with a security requirements approach. The order in which they are performed depends on the particular process that is established in the organization. Hence the sub-process and its activities described here can be combined with existing development methods such as the Unified Process or other development processes. In this paper we will describe the integration of our process into the SPL engineering framework proposed in [30].

As it is described in Fig. 1, SREPPLine is composed of two sub-processes with their respective activities: PLSecDomReq (Product Line Security Domain Requirements Engineering sub-process) and PLSecAppReq (Product Line Security Application Requirements Engineering sub-process). These sub-processes cover the four basic phases of the requirements engineering according to [20]: requirements elicitation; requirements analysis and negotiation; requirements documentation; and requirements validation and verification. Therefore, at least, they have to be performed for each iteration of the Domain or Application Requirements Engineering Process of the SPL respectively. However, in this paper due to space restrictions, we will only outline the key activities and tasks that have to be part of PLSecDomReq and we will present a brief overview of PLSecAppReq (as an evolution of SREP [28]).

3.1. Product line security domain requirements engineering sub-process

The main goals of this sub-process (SP1) are, firstly the development of common and variable security requirements which conform to IEEE 830:1998, secondly, their precise documentation in a Protection Profile[2] (PP) adapted document by following the standard ISO/IEC 15446 [10] and finally the development of their common and variable related security artefacts following a CC format. The process model of this sub-process will be shown in Fig. 2.

3.1.1. SP1—activity A1.1: security management scoping

This activity comprises the following tasks: security core assets repository improvement (up-to-date artefacts and links); identification of specific stakeholders; security definitions agreement; security environment identification (security policy, security standards, laws,

[1] A feature is an end-user visible characteristic of a system.

[2] The Common Criteria define a Protection Profile as an implementation independent statement of security requirements that is shown to address threats that exist in a specified environment.

:OMPUTER STANDARDS & INTERFACES
'he International Journal on the Development and Application of Standards for Computers, Software)uality, Data Communications, E-topics, Interfaces and Measurement

'ublication information

:omputer Standards & Interfaces (ISSN 0920-5489). For 2008 volume 30 (6 issues) is scheduled for ublication. Subscription prices are available upon request from the Publisher or from the Regional ales Office nearest you or from this journal's website (http://www.elsevier.com/locate/csi). Further iformation is available on this journal and other Elsevier products through Elsevier's website: ittp://www.elsevier.com). Subscriptions are accepted on a prepaid basis only and are entered on a alendar year basis. Issues are sent by standard mail (surface within Europe, air delivery outside urope). Priority rates are available upon request. Claims for missing issues should be made within six onths of the date of dispatch.

rders, claims, and product enquiries

ease contact the Customer Service Department at the Regional Sales Office nearest you:
rlando: Elsevier, Customer Service Department, 6277 Sea Harbor Drive, Orlando, FL 32887-4800, SA; phone: (+1) (877) 8397126 [toll free number for US customers], or (+1) (407) 3454020 [customers itside US]; fax: (+1) (407) 3631354; or (+1) (407) 3639661; e-mail: usjcs@elsevier.com or spcs@elsevier.com
nsterdam: Elsevier, Customer Service Department, PO Box 211, 1000 AE Amsterdam, The etherlands; Tel.: +31-20- 4853757; Fax: +31 20 4853432; E-mail: jp.info@elsevier.com
kyo: Elsevier, Customer Service Department, 9-15 Higashi-Azabu 1-chome, Minato-ku, Tokyo 106-)44, Japan; Tel.: +81- 3-55615033; Fax: +81-3-55615047; E-mail: jp.info@elsevier.com
ngapore: Elsevier, Customer Service Department, 3 Killiney Road, #08-01 Winsland House I, ngapore 239519; Tel.: +65-6349 0222; Fax: +65-6733 1510; E-mail: asiainfo@elsevier.com.sg
o de Janeiro: Elsevier, Rua Sete de Setembro 111/16 Andar, 20050-002 Centro, Rio de Janeiro - I, Brazil; Tel.: +55-21-509-5340; Fax: +55-21-507-1991; E-mail: elsevier@campus.com.br [Note (Latin nerica): for orders, claims and help desk information, please contact the Regional Sales Office in New rk as listed above]

lvertising information

vertising orders and enquiries can be sent to: **USA, Canada and South America:** Mr Tino DeCarlo, e Advertising Department, Elsevier Inc., 360 Park Avenue South, New York, NY 10010-1710, USA; .: +1-212-6333815; Fax: +1-212-6333820; E-mail: t.decarlo@elsevier.com. **Europe and ROW:** mmercial Sales Department, Elsevier Ltd., The Boulevard, Langford Lane, Kidlington, Oxford OX5 B, UK; Tel.: +44-1865-843016; Fax: +44-1865-843976; E-mail: media@elsevier.com. **Japan:** The vertising Department, Elsevier K.K., 9-15 Higashi-Azabu 1-chome, Minato-ku, Tokyo 106-0044,)an; Tel.: +81-3-55615033; Fax: +81-3-55615047.

thor enquiries. For enquiries relating to the submission of articles (including electronic submission ere available) please visit this journals homepage at http://www.elsevier.com/locate.csi. You can track :epted articles at http://www.elsevier.com/trackarticle and setup e-mail alerts to inform you of when an cle's status has changed. Also accessible from here is information on copyright, frequently asked :stions and more.

glish language help service: Authors who require information about language editing and copy-ting services pre- and post-submission please visit http://www.elsevier.com/locate/languagepolishing :ontact authorsupport@elsevier.com for more information. Please note Elsevier neither endorses nor :s any responsibility for any products, goods or services offered by outside vendors through our service ı any advertising. For more information please refer to our Terms & Conditions http://www.elsevier.com/ nsandconditions.

A mailing notice: *Computer Standards and Interfaces* (ISSN 0920-5489) is published bi-monthly Elsevier (P.O. Box 211, 1000 AE Amsterdam, The Netherlands). Annual subscription price in the USA $1,108 (valid in North, Central and South America), including air speed delivery. Periodical postage d at Rahway NJ and additional mailing offices.
A POSTMASTER: Send change of address to: *Computer Standards and Interfaces*, Elsevier, 6277 ı Harbor Drive, Orlando, FL 32887-4800.
FREIGHT AND MAILING in USA by Mercury International Limited, 365, Blair Road, Avenel,)7001.

ELSEVIER

Volume 30, Issue 6, August 2008

COMPUTER STANDARDS & INTERFACES

www.elsevier.com/locate/csi

CONTENTS

Abstracted/indexed in: INSPEC, Pascal, Ei Compendex, UnCover, SCISEARCH, Social SciSearch, Inside Conferences, Information Science & Technology Abstracts. Also covered in the abstract and citation database SCOPUS®. Full text available on ScienceDirect®.

0920-5489(200808)30:6;1-F

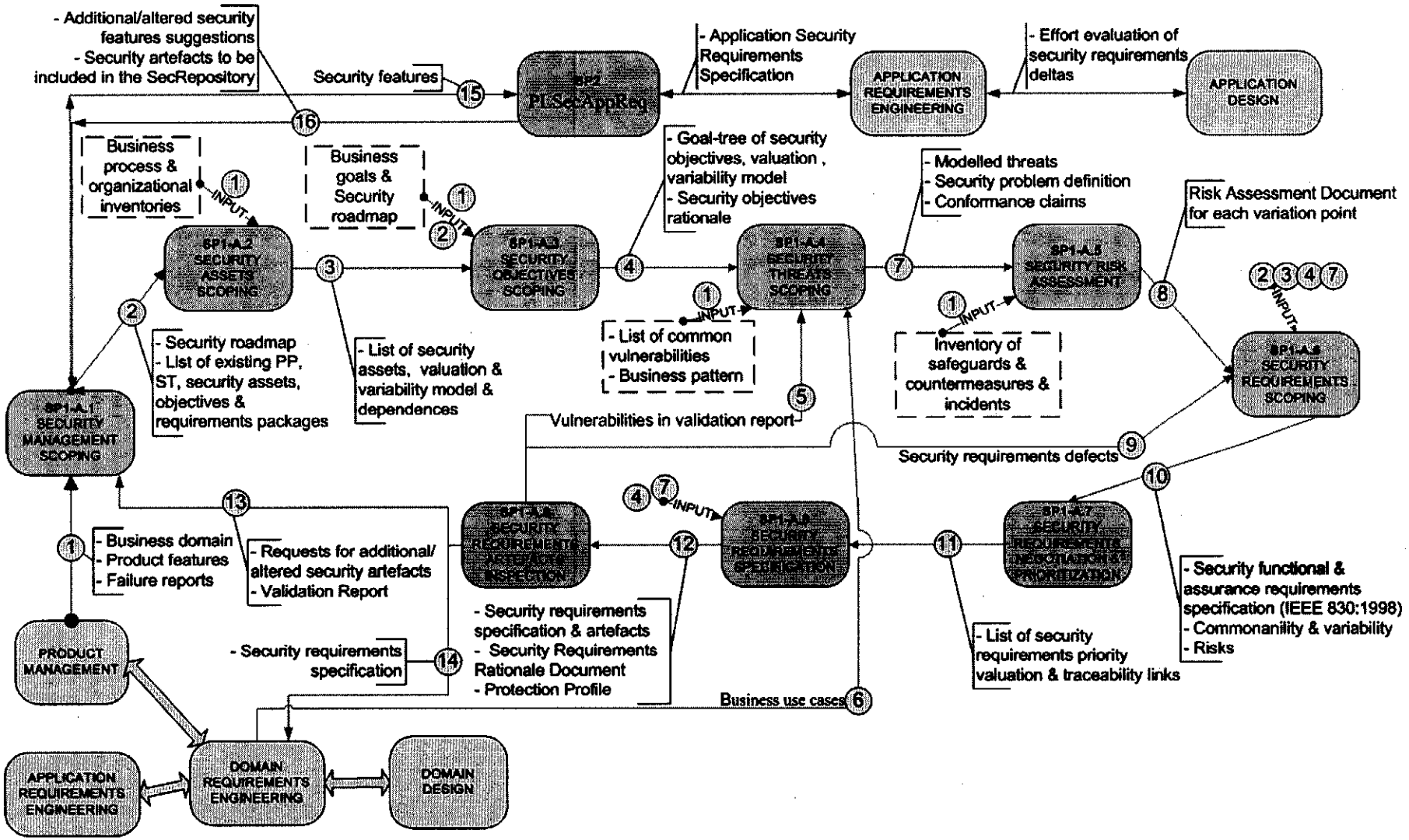

Fig. 2. Activity model of PLSecDomReq.

constraints, security needs and security acceptance criteria as well as the evaluation assurance level (EAL) of the CC); identification of the relevant type/categories of assets and security objectives; security features identification (commonalities and variability); and security cost impact and risk superficial estimations.

3.1.2. SP1—activity A1.2: security assets scoping

This activity comprises the following tasks: security assets identification for each asset (or group of them) and for the environment; security assumptions; security asset scoping, which aims at identifying particular components to be developed for reuse, common and variable assets; and identification of dependences between security assets.

3.1.3. SP1—activity A1.3: security objectives scoping

This activity comprises the following tasks: security objectives identification (commonality and variability analysis) for each asset; security objectives modelling and specification in XML (APE_OBJ CC class); and assets valuation against their related security objectives.

3.1.4. SP1—activity A1.4: security threats scoping

This activity comprises the following tasks: identification of potential vulnerabilities in public domain sources; identification of the attack tree associated with the business pattern or SPL domain; identification of the misuse cases and threats for each security objective and asset (commonality and variability analysis), because each asset is targeted by threat/s that can prevent the security objective from being achieved; threats modelling and specification; validation of security objectives against threats and assets and their variability models.

3.1.5. SP1—activity A1.5: security risks assessment

This activity comprises the following tasks in order to achieve 100% risk acceptance: assessing whether the threats are relevant according to the security level specified by the security objectives; estimating the security risks based on the relevant threats, their likelihood and their potential negative impacts, depending on the variation points. To do so, the ISO/IEC 13335 (GMITS) [9], provides guidance on the use of the risk management process. In Spain we could use MAGERIT [22], which conforms to ISO/IEC 13335 and soon to ISO/IEC 27005.

3.1.6. SP1—activity A1.6: security requirements scoping

This activity comprises the following tasks: security requirements elicitation (the appropriate CC security functional requirements and ISO/IEC 27001 control objectives should be also selected), the suitable security requirements or the suitable package of security requirements that mitigate the threats at the necessary levels with regard to the risk assessment must be selected; identification of common security requirements according to the elicited requirements and through the previously performed risk analysis; defining variable requirements and variability dependencies; security requirements modelling; definition of the allowed CC operations (iteration, assignment, selection or refinement); definition of security test/metric and countermeasure for each security requirement. Thus, at the end of this activity and according to ISO/IEC 17799:2005 the functional, assurance, and organizational security requirements, along with the security requirements for the IT development and operational environment must have been specified.

3.1.7. SP1—activity A1.7: security requirements negotiation and prioritization

This activity comprises the following tasks: interdependences with other functional and non-functional requirements and trade-offs in the security variability model and therefore in the orthogonal variability model; balancing the risk with the economical impact of implementing countermeasures.

3.1.8. SP1—activity A1.8: security requirements specification

This activity comprises the following tasks: security requirements modelling and security requirements specification.

3.1.9. SP1—activity A1.9: security requirements artefacts inspection

This activity comprises the following tasks: i) it is verified whether the security requirements conform to ISO/IEC 27001 control objectives and to CC (ISO/IEC 15408) assurance requirements and to the IEEE 830-1998 standard, because according to this standard, a quality requirement has to be correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable, and traceable. ii) Furthermore, PLSecAppReq proposes the use of the SSE-CMM (ISO/IEC 21827) in order to help in the evaluation of the product line security engineering process in the Domain Testing sub-process, with the help of the CC_SSE-CMM [21] approach. Thereby, we propose to evaluate the security of the SPL along with the product line security engineering process by using the CC assurance requirements and the SSE-CMM at the same time with the help of CC_SSE-CMM. iii) It is verified in the Domain Testing sub-process the fulfilment of the previously approved EAL and the CC evaluation.

3.2. Product line security application requirements engineering sub-process

PLSecAppReq (Product Line Security Application Requirements Engineering sub-process) activities in this sub-process (SP2) are the evolution of the activities proposed in SREP [28], together with some new specific tasks and activities that are unique to security requirements engineering in SPL. The main goals of this sub-process are: elicitation and documentation of the security requirements and their related security artefacts; to make them conform to IEEE 830:1998, as well as to gather them in a Security Targets[3] (ST) adapted document by following the ISO/IEC 15446 [10] standard; at the same time to reuse, as much as possible, the security domain artefacts and requirements.

Due to the fact that this sub-process is not one of the targets of study in this article, we have enumerated in Fig. 1 the key activities of PLSecAppReq sub-process and below we will only describe the key high-level concepts that make this sub-process different from security requirements engineering processes for single systems (such as SREP):

- Requirements elicitation is based on the communication of the available commonality and variability of the SPL. Most of the requirements are not elicited anew, but are derived from the security domain requirements.
- During security requirements elicitation, the differences between security application artefacts and security domain artefacts (secdeltas) must be detected, evaluated with regard to the required adaptation effort, and suitably documented. If the required adaptation effort is early known, trade-off decisions concerning application security artefacts are possible not only to reduce the effort but also to increase the amount of domain artefacts.

4. Security core assets repository

SREPPLine proposes a Security Core Assets Repository (SCAR), which facilitates product line security requirements engineering by making it easier the development with security requirements reuse, which is an important concept in SPL because it helps us increase the security requirements quality for an improved use in subsequent projects [35]. Moreover, it helps manage in an easier way one of the most important factors of success in the development of a secure SPL: the management of commonalities and variability of the security requirements and their traceability links.

[3] The Common Criteria define a Security Target as an implementation dependent statement of security needs for a specific identified product.

Security requirements can be obtained from the assets starting from the features for a new SPL or for a new product of a SPL. Furthermore, the SCAR uses the concept of Sec-Domain Requirements package, understood as a homogeneous set of security requirements and their related security artefacts that can be applied to different SPL, and that are put together to satisfy the same security objectives as well as to mitigate the same threats, being a larger and more effective reuse unit (such as a personal data legislation conformance package). The security domain requirements engineering as well as the security application requirements engineering sub-processes are supported by this repository.

Next, we will outline the most important and/or complex aspects of the meta-model shown in Fig. 3:

- 'Threat' and 'Security Requirement' can be represented as different specifications, thanks to the elements 'Threat Specification' and 'Security Requirement Specification'. They are included in the core assets of the SPL, together with the 'Security Objective', which is documented through a goal tree. In addition, each product of the SPL derives these elements and refines them in order to describe the particularities of the product properly.
- The 'Security Requirement–Security Requirement' relationship allows an inclusive or exclusive trace between requirements, that is variability constraints. An exclusive trace between requirements means that they are mutually alternative, for example that they are in conflict or overlapping, whereas, an inclusive trace between requirements means that to satisfy one, other/other/s is/are needed to be satisfied. In addition, this repository is used to model the dependencies with other functional and non-functional requirements. Traces between requirements in the SPL infrastructure do automatically also exist in the products.

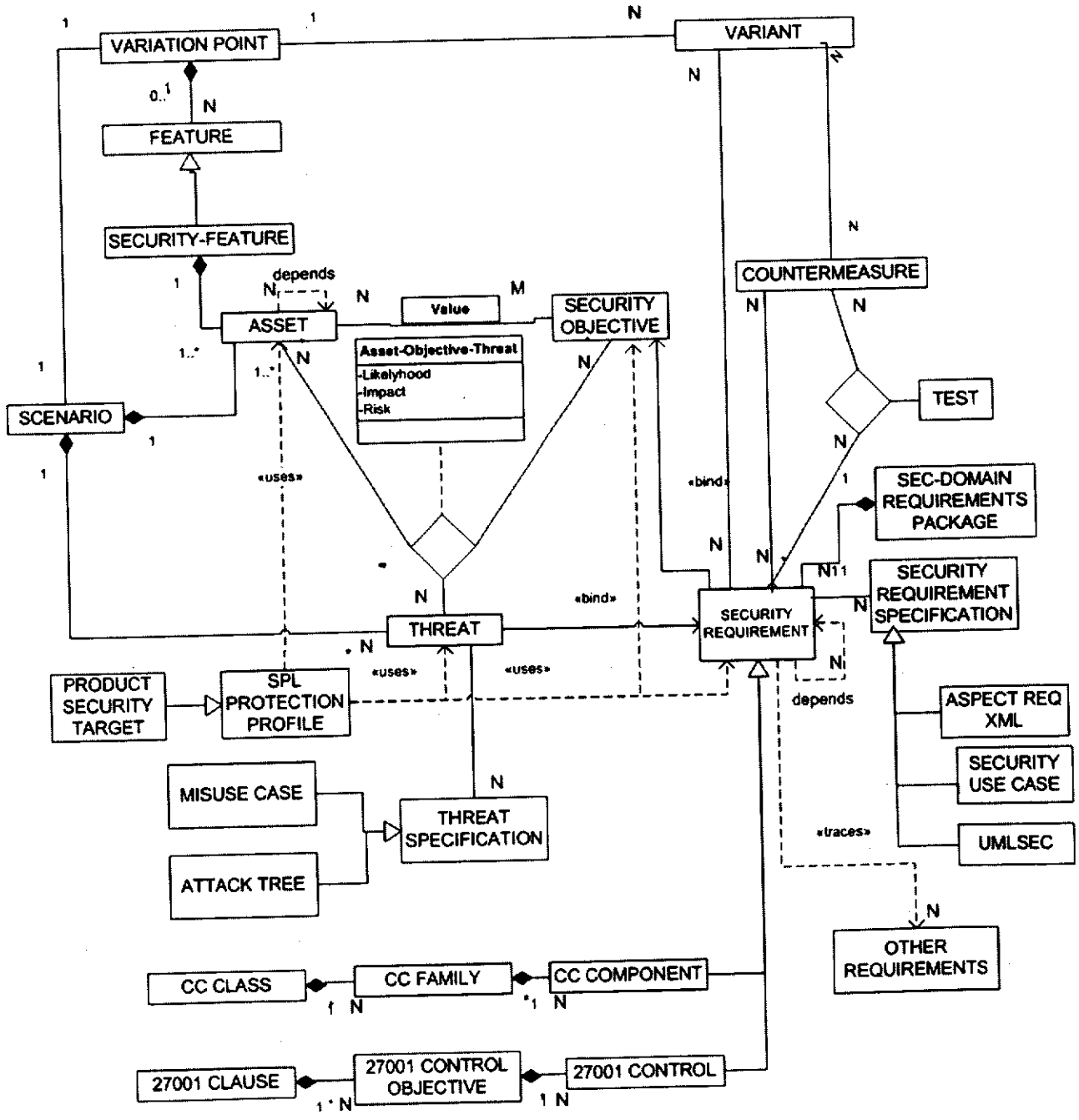

Fig. 3. Security core assets repository meta-model or security requirements decision model.

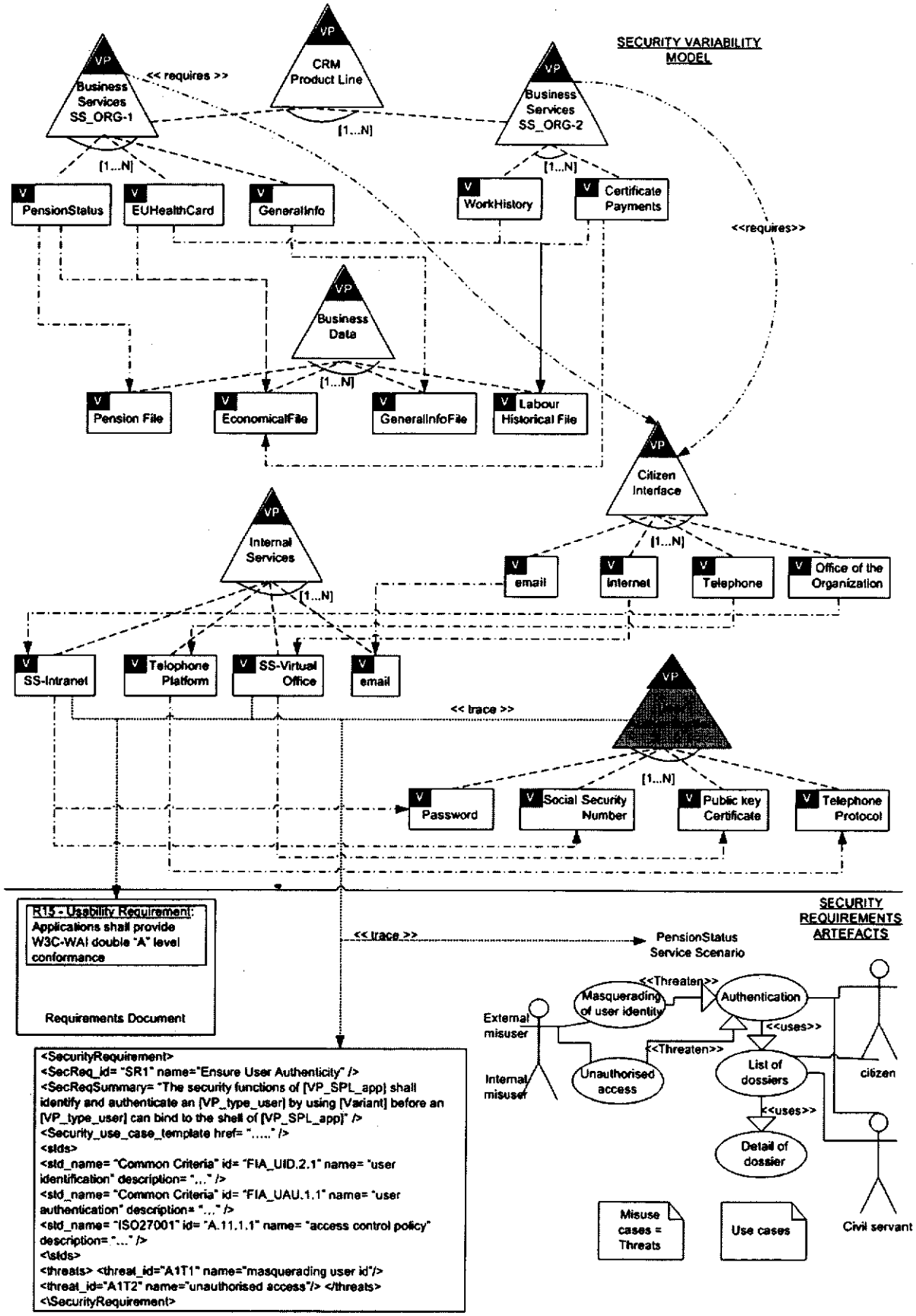

Fig. 4. Example: CRM security variability model and security requirements artifacts.

In addition, there could have been links further on to design level specifications, security test cases, countermeasures, etc., as well as traceability links with other artefacts ("artefacts dependencies"), through the relationship between a variant of the orthogonal variability model and the associated development artefact, because our proposed model process is based on the concept of the orthogonal variability model of Pohl et al. [30]. Thus, the SCAR must be integrated into the SPL core assets repository to facilitate these traceability links between the variability model of the SPL and the different types of security artefacts and the other development artefacts. XML is used for specifying and managing security requirements and other artefacts of the family in order to enable the explicit documentation of variability in a structured format and to facilitate the traceability links. In fact, it is an aspect-oriented security requirements specification based on XML.

Finally, we would like to point out the fact that using the CC, a large number of security requirements on the SPL itself and on the products development can be defined. Nevertheless, the CC neither provide us with methodological support, nor contain security evaluation criteria pertaining to administrative security measures not directly related to IS security measures. However, it is known that an important part of the security of an IS can be often achieved through administrative measures. Therefore, according to ISO/IEC 17799:2005, we propose to include legal, statutory, regulatory, and contractual requirements that the organization, its trading partners, contractors, and service providers have to satisfy, and their socio-cultural environment. After converting these requirements into software and system requirements format, they along with the CC security requirements and the ISO/IEC 27001 control objectives would be the initial subset of security requirements of a SPL, acting like horizontal security requirements patterns for the software product lines of the organization.

5. A SREPPLine application scenario

In this section, we will describe how the sub-process of SREPPLine, PLSecDomReq (Product Line Security Domain Requirements Engineering), can be applied in practice.

We will apply our process (PLSecDomReq) to specify the security requirements of a software product line of a CRM (Customer Relationship Management) system, which may have several different configurations for three different public institutions of the public social security system of Spain. Therefore, we will characterize the system, named SS-CRM, as a SPL whose members vary by system configuration yet retain the same core functionalities. Obviously, this limited scope of variability is not representative of most product lines. However, it would be an instructive exercise to apply our process in a real scenario; taking into account the fact that this case study has to be simplified and summed up to enable points of PLSecDomReq to be easily illustrated in this article.

The IT Department of the Organization in charge of the development of the SPL had already loaded into the Security Core Assets Repository a SPL basic profile, with the most regular security artefacts for the current products of the Organization, such as legal, regulatory and policy norms, together with the CC security requirements and the ISO/IEC 27001 controls.

5.1. SP1—activity A1.1: security management scoping

As an input of this activity we received the feature model tree of SS-CRM, in which the functional and domain components variability are described. From this feature model we identified the security features and their dependences and we developed the security orthogonal variability model and we specified it in XML. In Fig. 4, part of the security variability model is shown, it is represented the basic features of the SPL together with their related assets as well as it is represented the security feature of user authentication and its relations with the different assets depending on the variants.

We identified the following business security objectives or security dimensions [22]: integrity (I), confidentiality (C), availability (D), authenticity of service users (A_S), authenticity of data origin (A_D), accountability (or traceability) of service use (T_S) and accountability of data access (T_D). Furthermore, we identified the following categories or types of assets, as shown in Table 1, after analysing the security policy of the Organization, the business processes, business use cases and SPL environment: business services and business data, internal services and equipment (hardware, software and communications), although we will not take into account the last type of assets in order to simplify the application of SREPPLine.

We also reached an agreement upon several security concepts definition and upon the security level needed (we stated that the SPL and their products have to conform to EAL 2 of the CC at least), as well as we decided that due to the type of information managed by SS-CRM, it would have to fulfil the Spanish privacy data protection legislation.

5.2. SP1—activity A1.2: security assets scoping

In this activity we identified the common and variable security assets for each security feature; and the dependences between them. In Table 1 part of the security assets are listed and categorized by security feature. In addition, in Fig. 4 these dependences are represented in the security variability model. Moreover, we were helped by the SCAR in the task of security assets identification and categorization, so that if the security features identified in the previous activity were in the SCAR we would be able to retrieve their associate security assets. (In the following activities SCAR could be used in the same way for the identification of the rest of artefacts: security objectives, threats and security requirements).

5.3. SP1—activity A1.3: security objectives scoping

In this activity from the security dimensions identified in activity A1.1, we stated the security objectives (together with a commonality and variability analysis) for each security asset as well as the security assets valuation against their related security objectives. In order to carry out this task we performed interviews to the different stakeholders along with the Delphi evaluation method and the value scale proposed in MAGERIT [22] (from 0 to 10), because it would be the risk assessment method that we would use next. Thus, following the qualitative model of MAGERIT only the higher assets in the dependencies tree obtained in the previous activity were explicitly valued. Then, when this valuation was propagated through the dependencies tree of the variability model, we obtained the table of accumulated value for each of the identified assets in the SPL. Part of this table of accumulated value for each asset is shown in Table 1. In this table, the first value of each cell is the value of the asset, and if the number is between brackets it is a propagated value. Finally, we specified in XML these security objectives together with their values for each asset.

5.4. SP1—activity A1.4: security threats scoping

As an input of this activity we received the list of the most common vulnerabilities and threat / attack patterns for the Organization as well as the catalogue of threats listed in the Organization. With all these data, together with the help of the SCAR and after analysing the business use cases and developing the misuse cases, identifying the potential misusers at the same time, we identified the common threats for each asset and we specified them with misuse cases templates and we traced them to the security variability model. Some of these threats are listed in Table 1, where the wilful threats that threaten the asset "Business Service Pension Status" are shown. In addition, an example

Table 1
Part of the risk assessment map

(A) Assets (T) Threats	Frequency	Security objectives / security dimensions [D]	[I]	[C]	[A_S]	[A_D]	[T_S]	[T_D]
[BS] Business Services								
(A) [BS_PensionStatus] Status citizen's pension		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
(T) Manipulation of configuration	0,1	50%; 4; 2			100%; 7; 4		100%; 6; 3	
(T) Masquerading of user identity	100				100%; 7; 5			
(T) Misuse	10	70%; 5; 4			10%; 4; 4		50%; 5; 4	
(T) Re-routing of messages	10				50%; 6; 5		50%; 5; 4	
(T) Unauthorised access	100	10%; 2; 3			50%; 6; 5			
(T) Repudation	10						100%; 6; 5	
(T) Denial of Service	10	50%; 4; 4						
(A) [BS_EUHealthcareCard] EU Healthcare Card		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
(A) [BS_GeneralInfo] General Information about SS		5; 70%; 5; 4			1; 100%; 1; 3		1; 100%; 1; 2	
(A) [BS_WorkHistory] Certified Work History		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
(A) [BS_CertificatePayments] up-to-date SS payments		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
[BD] Business Data								
(A) [D_Pension] Pension Files		[5]; 90%; 5; 5	5; 50%; 4; 4	7; 100%; 7; 5	[7]; 100%; 7; 5	7; 100%; 7; 4	[6]; 100%; 6; 3	5; 100%; 5; 3
(A) [D_SS_Economical] Economical Files SS		[5]; 90%; 5; 5	5; 50%; 4; 4	6; 100%; 6; 5	[7]; 100%; 7; 5	6; 100%; 6; 3	[6]; 100%; 6; 3	5; 100%; 5; 3
(A) [D_GeneralInfo] General Information		[5]; 90%; 5; 5	3; 50%; 2; 3	0; 100%; 0; 1	[1]; 100%; 1; 2	2; 100%; 2; 1	[1]; 100%; 1; 1	1; 100%; 1; 1
(A) [D_Labour] Labour historical file		[5]; 90%; 5; 5	5; 50%; 4; 4	6; 100%; 6; 5	[7]; 100%; 7; 5	6; 100%; 6; 4	[6]; 100%; 6; 3	5; 100%; 5; 3
[IS] Internal Services								
(A) [IS_email] email		[5]; 70%; 5; 4	[5]; 50%; 2; 3	[0]; 50%; 0; 0	[1]; 100%; 1; 3	[2]; 100%; 2; 3	[1]; 100%; 1; 2	[1]; 100%; 1; 1
(A) [IS_Telephone] Telephone		[5]; 70%; 5; 4	[5]; 50%; 4; 5	[6]; 50%; 5; 5	[7]; 100%; 7; 5	[6]; 100%; 6; 5	[6]; 100%; 6; 5	[5]; 100%; 5; 4
(A) [IS_VirtualOffice] Virtual Office of SS		[5]; 70%; 5; 4	[5]; 50%; 4; 5	[7]; 50%; 6; 5	[7]; 100%; 7; 5	[7]; 100%; 7; 5	[6]; 100%; 6; 5	[5]; 100%; 5; 4
(A) [IS_Intranet] Intranet for CRM civil servants		[5]; 70%; 5; 4	[5]; 50%; 4; 5	[7]; 50%; 6; 5	[7]; 100%; 7; 5	[7]; 100%; 7; 5	[6]; 100%; 6; 5	[5]; 100%; 5; 4

scenario of the misuse cases will be shown in Fig. 4, where some of the misuse cases for the asset "Business Service Pension Status" are represented and traced to the security variability model.

5.5. SP1—activity A1.5: security risks assessment

This activity comprises risk assessment, so that we used MAGERIT [22]. Therefore, first of all we estimated the likelihood (in terms of frequency of occurrence from 0 to 100: 100 for very frequent, daily; 10 for frequent, monthly; 1 for normal, annually; 0'1 for infrequent, every few years) for each threat to the assets, and the degradation of the assets expressed as a percentage of their value, with the help of the SCAR and historical information of the Organization. Then, the accumulated impact on the assets was estimated by taking into account the accumulated value on the assets and the degradation value caused by the threat. Next, the accumulated risk to the assets was estimated by taking into account the accumulated impact and the estimated frequency of occurrence of the threat. In Table 1 part of the accumulated impact estimation is shown as well as part of the risk assessment is presented (the risk is classified in a range from 0, almost null; 1–2 for low risk; 3 for medium risk; 4 for high risk; to 5, for very high risk). In Table 1, the second number of each cell is the degradation value of the assets caused by the threat expressed as a percentage; the third value is the accumulated impact to the assets; ant the last (fourth) value is the accumulated risk to the assets.

5.6. SP1—activity A1.6: security requirements scoping

In this activity, first of all we analysed the misuse cases and their related threats, then we selected the appropriate CC security functional requirements and ISO/IEC 27001 controls for the threats of the product line. After that, we performed the identification of the common security requirements according to the elicited requirements and through the previously performed risk analysis, defining variable security requirements and variability dependencies between them, at the same time when it was a CC security requirement we defined the allowed CC operations (iteration, assignment, selection or refinement). Finally we modelled the security requirements with security use cases; and we traced them with their associated security test and security measure/metric. In Fig. 4 an example of some security use cases for the asset "Business Service Pension Status" are shown.

5.7. SP1—activity A1.7: security requirements negotiation and prioritization

In this activity the security requirements were prioritized according to the risk. Then we identified and specified in our security variability model the interdependences of the security requirements with other functional and non-functional requirements by analysing the use cases and the feature model (in Fig. 4 an example of an interdependence between a security requirement and another type of requirement is represented). Besides we performed a slight economical analysis balancing the risk with the economical impact of implementing countermeasures. Thereby we reached an agreement with the stakeholders about taking into account those security requirements associated to those threats that imply high or very high risk whatever the conflicts with other requirements. However, for the security requirements with less risk than high (that is, from 3 to 1, medium to low) we had to reach trade-offs with other non-functional requirements mainly, especially about performance and interface accessibility (for example, the system had to fulfil the WAI, Web Accessibility Initiative, level 'AA').

5.8. SP1—activity A1.8: security requirements specification

This activity comprised security requirements modelling and security requirements specification. In order to do so, we used the technique of security use cases and their parametrical templates, which we traced to the security variability model. In Fig. 4 we can see an example of part of a security requirement specification with the technique of XML aspect requirements specification, and with its traces to the security variability model and to the security use case template.

5.9. SP1—activity A1.9: security requirements artefacts inspection

In this activity we verified whether the security requirements conform to ISO/IEC 27001 control objectives and CC (ISO/IEC 15408) assurance requirements of the EAL2 and the IEEE 830-1998 standard. Moreover, we estimate the residual risk of the SPL in order to evaluate the effectiveness of the security requirements and their countermeasures (following the Plan-Do-Check-Act process model).

6. Conclusions and further work

Nowadays, software security is generating a growing interest mainly due to the increasingly crucial nature of the IS with corresponding levels of new legal and governmental requirements. At the same time, there is an increasing need to obtain high-quality IS along with higher productivity. For this reason, software product line based development has become the most successful approach for ensuring quality, economic efficiency and manageability of software systems [2]. Nevertheless, requirements management and security of information are much more critical in SPL due to the complexity and extensive nature of them. Therefore, we believe that it is vital to deal with security at all stages of SPL development, especially in the management of security requirements, since they form the basis for the achievement of a robust IS.

Hence, the contribution of this work is that of providing a security standard-based process that deals with security requirements from the early stages of SPL development in a systematic and intuitive way specially adapted for SPL based development, which is based on the use of the latest security requirements techniques, such as UMLSec [14], security use cases [5] or misuse cases [33]; as well as on the reuse of security artefacts, by providing a Security Core Assets Repository (SCAR), together with the integration of the Common Criteria (ISO/IEC 15408) and ISO/IEC 27001 controls into the SPL lifecycle. Moreover, it also conforms to the most relevant security standards with regard to the management of security requirements such as the aforementioned ISO/IEC 15408 and ISO/IEC 17799:2005 (sections: 0.3, 0.4, 0.6 and 12.1) and ISO/IEC 27001 standard (sections: 4.2.1, 4.2.3, 4.3, 6.a, 6.b and A.12.1.1). Furthermore, it facilitates that the products of the SPL conform to these former standards as well as the fulfilment of the IEEE 830:1998 standard. In addition, SREPPLine suggests using a method to carry out the risk assessment which conforms to ISO/IEC 13335 (GMITS).

Among the most important lessons learned from the case study presented above we can highlight the following ones:

- The application of this case study has allowed us to improve and refine several activities of SREPPLine.
- Tool support is critical for the practical application of this process to large-scale software systems due to the number of handled artifacts and the complexity of the traceability relations and the variability model.
- With respect to the benefits obtained by the Organization in which the case study was carried out, it has managed to have normalized a systematic and specific process for the management of security requirements in SPL which conforms to ISO/IEC 15408 and ISO/IEC 27001, as well as the creation of a security core assets repository whose artefacts will be reused for the development of the products of the SPL and also they could be reused for the development of future SPL in the Organization.

Therefore, further work is needed to develop a new version of our previously developed CARE (Computer-Aided Requirements Engineering) tool (SREPTOOL [26]) in order to support and gather the new characteristics of our proposed security requirements engineering process for software product lines. Furthermore, we will carry out a refinement of the theoretical approach by proving it with more real and complete case studies to complete and deeply detail SREPPLine. Finally, we will work to complement this process in order to provide a holistic framework for security engineering in software product lines.

Acknowledgements

This paper is part of the ESFINGE (TIN2006-15175-C05-05) and RETISTRUST (TIN2006-26885- E) projects of the Ministry of Education and Science (Spain), and of the MISTICO (PBC-06-0082) and DIMENSIONS (PBC-05-012-2) projects of the Consejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla- La Mancha and the FEDER.

References

[1] R. Baskeville, The development duality of information systems security, Journal of Management Systems 4 (1) (1992) 1–12.
[2] A. Birk, G. Heller, I. John, T.v.d. MaBen, K. Müller, K. Schmid, Product Line Engineering Industrial Nuts and Bolts, Fraunhofer IESE, Kaiserslautern, 2003.
[3] J. Bosh, Design & Use of Software Architectures, Pearson Education Limited, 2000.
[4] P. Clements, L. Northrop, Software Product Lines: Practices and Patterns. SEI Series in Software Engineering, Addison-Wesley, 2002.
[5] D.G. Firesmith, Engineering security requirements, Journal of Object Technology 2 (1) (2003) 53–68.
[6] D.G. Firesmith, Security use cases, Journal of Object Technology (2003) 53–64.
[7] IEEE, IEEE 830: 1998 Recommended Practice for Software Requirements Specifications, 1998.
[8] ISO/IEC, ISO/IEC 21827:2002 Information Technology – Systems Security Engineering – Capability Maturity Model (SSE-CMM), 2002.
[9] ISO/IEC, ISO/IEC 13335 Information technology – Security Techniques – Management of information and Communications Technology Security – Part 1: Concepts and Models for Information and Communications Technology Security Management, 2004.
[10] ISO/IEC, ISO/IEC 15446 Information Technology – Security Techniques – Guide for the Production of Protection Profiles and Security Targets, 2004.
[11] ISO/IEC, ISO/IEC 15408:2005 Information technology – Security Techniques – Evaluation Criteria for IT Security, (Common Criteria v3.0), 2005.
[12] ISO/IEC, ISO/IEC 17799 Information Technology – Security Techniques – Code of Practice for Information Security Management, 2005.
[13] ISO/IEC, ISO/IEC 27001:2005 Information Technology – Security Techniques – Information Security Management Systems – Requirements, 2005.
[14] J. Jürjens, UMLsec: extending UML for secure systems development. UML 2002—the unified modeling language. model engineering, languages, concepts, and tools, 5th International Conference, LNCS, vol. 2460, 2002, pp. 412–425.
[15] T. Käkölä, J.C. Dueñas, Software Product Lines: Research Issues in Engineering and Management, Springer, 2006.
[16] K. Kang, S. Cohen, J.A. Hess, W.E. Novak, S.A. Peterson, Feature-oriented Domain Analysis (FODA) Feasibility Study, Software Engineering Institute, Carnegie-Mellon University, 1990.
[17] J. Kim, M. Kim, S. Park, Goal and scenario bases domain requirements analysis environment, Journal of Systems and Software 79 (7) (2005) 926–938.
[18] H.K. Kim, Automatic translation form requirements model into use cases modeling on UML ICCSA 2005, LNCS, 2005: p. 769–777.
[19] G. Kotonya, I. Sommerville, Requirements Engineering Process and Techniques, John Willey & Sons, UK, 1998, p. 294, Hardcover ed.
[20] G. Kotonya, I. Sommerville, Requirements Engineering Process and Techniques, John Willey & Sons, 2000.
[21] J. Lee, J. Lee, S. Lee, B. Choi, A CC-based security engineering process evaluation model, 27th Annual International Computer Software and Applications Conference (COMPSAC'03), 2003, p. 130.
[22] F. López, M.A. Amutio, J. Candau, J.A. Mañas, Methodology for Information Systems Risk Analysis and Management, Ministry of Public Administration, 2005.
[23] F. Massacci, M. Prest, N. Zannone, Using a security requirements engineering methodology in practice: the compliance with the Italian data protection legislation, Computers Standards and Interfaces 27 (2005) 445–455.
[24] J. McDermott, C. Fox, Using abuse case models for security requirements analysis, Annual Computer Security Applications Conference, Phoenix, Arizona, 1999.
[25] N.R. Mead, Identifying Security Requirements Using the Security Quality Requirements Engineering (SQUARE) method, in: H. Mouratidis, P. Giorgini (Eds.), Integrating Security and Software Engineering: Advances and Future Visions, Idea Group Publishing, 2007.

[26] Mellado, D., Fernández-Medina, E., Pattini, M., Automated Support for Security Requirements Engineering in Practice. XXXIII Latinamerican Conference of Computer Science (CLEI 2007), in press: p. 55.
[27] D. Mellado, E. Fernández-Medina, M. Piattini, A comparative study of proposals for establishing security requirements for the development of secure information systems, The 2006 International Conference on Computational Science and its Applications (ICCSA 2006), LNCS, vol. 3982, Springer, 2006, pp. 1044–1053.
[28] D. Mellado, E. Fernández-Medina, M. Piattini, A common criteria based security requirements engineering process for the development of secure information systems, Computer Standards and Interfaces 29 (2) (2007) 244–253.
[29] H. Mouratidis, P. Giorgini, Integrating Security and Software Engineering: Advances and Future Visions, Idea Group Publishing, 2007.
[30] K. Pohl, G. Böckle, F.v.d. Linden, Software Product Line Engineering. Foundations, Principles and Techniques, Springer, Berlin Heidelberg, 2005.
[31] G. Popp, J. Jürjens, G. Wimmel, R. Breu, Security-critical system development with extended use cases, 10th Asia-Pacific Software Engineering Conference, 2003, pp. 478–487.
[32] K. Schmid, K. Krennrich, M. Eisenbarth, Requirements Management for Product Lines: A Prototype, Fraunhofer IESE, 2005.
[33] G. Sindre, A.L. Opdahl, Eliciting security requirements with misuse cases, Requirements Engineering 10 (1) (2005) 34–44.
[34] M.T. Siponen, Secure-system design methods: evolution and future directions, IT Professional 8 (3) (2006) 40–44.
[35] A. Toval, J. Nicolás, B. Moros, F. García, Requirements reuse for improving information systems security: a practitioner's approach, Requirements Engineering 6 (4) (2002) 205–219.

Daniel Mellado has MSc in Computer Science from the Autonomous University of Madrid (Spain), and a PhD student at the Escuela Superior de Informática of the Castilla-La Mancha University (Spain). He is an Assistant Professor of the Department of Information Technologies and Systems at the Universidad de Castilla-La Mancha at Toledo (Spain). He participates at the ALARCOS research group of the Department of Information Technologies and Systems at the University of Castilla-La Mancha. He is a civil servant at the Social Security IT Department (in Madrid, Spain), where he works as an IT Project Manager. His research activities are security requirements engineering, security in information systems, secure software process improvement and auditory. He has several dozens of papers in national and international conferences on these subjects and co-author of several chapter books.

Eduardo Fernández-Medina has PhD and MSc in Computer Science. He is an Associate Professor at the Escuela Superior de Informática of the Universidad de Castilla-La Mancha at Ciudad Real (Spain). His research activities are security requirements, security in databases, data warehouses, web services and information systems, and also in security metrics. He is a co-editor of several books and chapter books on these subjects, and has several dozens of papers in national and international conferences. He participates at the ALARCOS research group of the Department of Information Technologies and Systems at the University of Castilla-La Mancha, in Ciudad Real (Spain). He belongs to various professional and research associations (ATI, AEC, AENOR, IFIP, WG11.3, etc).

Mario Piattini has MSc and PhD in Computer Science from the Politechnical University of Madrid. He is certified as an information system auditor by ISACA (Information System Audit and Control Association). He is an Associate Professor at the Escuela Superior de Informática of the Castilla-La Mancha University (Spain). He is an author of several books and papers on databases, security, software engineering and information systems. He leads the ALARCOS research group of the Department of Information Technologies and Systems at the University of Castilla-La Mancha, in Ciudad Real (Spain). His research interests are: advanced database design, database quality, software metrics, object-oriented metrics and software maintenance.

:OMPUTER STANDARDS & INTERFACES
'he International Journal on the Development and Application of Standards for Computers, Software)uality, Data Communications, E-topics, Interfaces and Measurement

'ublication information

:omputer Standards & Interfaces (ISSN 0920-5489). For 2008 volume 30 (6 issues) is scheduled for ublication. Subscription prices are available upon request from the Publisher or from the Regional ales Office nearest you or from this journal's website (http://www.elsevier.com/locate/csi). Further iformation is available on this journal and other Elsevier products through Elsevier's website: ittp://www.elsevier.com). Subscriptions are accepted on a prepaid basis only and are entered on a alendar year basis. Issues are sent by standard mail (surface within Europe, air delivery outside urope). Priority rates are available upon request. Claims for missing issues should be made within six onths of the date of dispatch.

rders, claims, and product enquiries

ease contact the Customer Service Department at the Regional Sales Office nearest you:
rlando: Elsevier, Customer Service Department, 6277 Sea Harbor Drive, Orlando, FL 32887-4800, SA; phone: (+1) (877) 8397126 [toll free number for US customers], or (+1) (407) 3454020 [customers itside US]; fax: (+1) (407) 3631354; or (+1) (407) 3639661; e-mail: usjcs@elsevier.com or spcs@elsevier.com
nsterdam: Elsevier, Customer Service Department, PO Box 211, 1000 AE Amsterdam, The etherlands; Tel.: +31-20- 4853757; Fax: +31 20 4853432; E-mail: jp.info@elsevier.com
kyo: Elsevier, Customer Service Department, 9-15 Higashi-Azabu 1-chome, Minato-ku, Tokyo 106-)44, Japan; Tel.: +81- 3-55615033; Fax: +81-3-55615047; E-mail: jp.info@elsevier.com
ngapore: Elsevier, Customer Service Department, 3 Killiney Road, #08-01 Winsland House I, ngapore 239519; Tel.: +65-6349 0222; Fax: +65-6733 1510; E-mail: asiainfo@elsevier.com.sg
o de Janeiro: Elsevier, Rua Sete de Setembro 111/16 Andar, 20050-002 Centro, Rio de Janeiro - I, Brazil; Tel.: +55-21-509-5340; Fax: +55-21-507-1991; E-mail: elsevier@campus.com.br [Note (Latin nerica): for orders, claims and help desk information, please contact the Regional Sales Office in New rk as listed above]

lvertising information

vertising orders and enquiries can be sent to: **USA, Canada and South America:** Mr Tino DeCarlo, e Advertising Department, Elsevier Inc., 360 Park Avenue South, New York, NY 10010-1710, USA; .: +1-212-6333815; Fax: +1-212-6333820; E-mail: t.decarlo@elsevier.com. **Europe and ROW:** mmercial Sales Department, Elsevier Ltd., The Boulevard, Langford Lane, Kidlington, Oxford OX5 B, UK; Tel.: +44-1865-843016; Fax: +44-1865-843976; E-mail: media@elsevier.com. **Japan:** The vertising Department, Elsevier K.K., 9-15 Higashi-Azabu 1-chome, Minato-ku, Tokyo 106-0044,)an; Tel.: +81-3-55615033; Fax: +81-3-55615047.

thor enquiries. For enquiries relating to the submission of articles (including electronic submission ere available) please visit this journals homepage at http://www.elsevier.com/locate.csi. You can track :epted articles at http://www.elsevier.com/trackarticle and setup e-mail alerts to inform you of when an cle's status has changed. Also accessible from here is information on copyright, frequently asked :stions and more.

glish language help service: Authors who require information about language editing and copy-ting services pre- and post-submission please visit http://www.elsevier.com/locate/languagepolishing :ontact authorsupport@elsevier.com for more information. Please note Elsevier neither endorses nor :s any responsibility for any products, goods or services offered by outside vendors through our service ı any advertising. For more information please refer to our Terms & Conditions http://www.elsevier.com/ nsandconditions.

A mailing notice: *Computer Standards and Interfaces* (ISSN 0920-5489) is published bi-monthly Elsevier (P.O. Box 211, 1000 AE Amsterdam, The Netherlands). Annual subscription price in the USA $1,108 (valid in North, Central and South America), including air speed delivery. Periodical postage d at Rahway NJ and additional mailing offices.
A POSTMASTER: Send change of address to: *Computer Standards and Interfaces*, Elsevier, 6277 ı Harbor Drive, Orlando, FL 32887-4800.
FREIGHT AND MAILING in USA by Mercury International Limited, 365, Blair Road, Avenel,)7001.

ELSEVIER

Volume 30, Issue 6, August 2008

COMPUTER STANDARDS & INTERFACES

www.elsevier.com/locate/csi

CONTENTS

Abstracted/indexed in: INSPEC, Pascal, Ei Compendex, UnCover, SCISEARCH, Social SciSearch, Inside Conferences, Information Science & Technology Abstracts. Also covered in the abstract and citation database SCOPUS®. Full text available on ScienceDirect®.

0920-5489(200808)30:6;1-F