ARES Conference

The International Dependability Conference

ARES 2009

16-19 March 2009

Fukuoka Institute of Technology
Fukuoka, Japan

Published by

IEEE computer society

CPS Conference Publishing Services

Message from ARES General Co-chairs

The Fourth International Conference on Availability, Reliability and Security (ARES 2009 – The International Dependability Conference) brings together researchers and practitioners in the area of dependability. ARES 2009 highlights the various aspects of dependability, with special focus on the crucial linkage between availability, reliability and security.

ARES aims at a full and detailed discussion of research issues in the field of dependability as an integrative concept that covers amongst others availability, safety, confidentiality, integrity, maintainability and security and their different areas of applications.

This conference emphasizes the interplay between foundations and practical issues of dependability in areas such as information systems, e-government, m-government, location-based services, ubiquitous computing, and autonomous computing.

This years ARES conference is devoted to establishing collaborations between different sub-disciplines and building a strong community for further research.

We are very happy to welcome three well-known keynote speakers:
- Elisa Bertino (Purdue University),
- Sushil Jajodia (George Mason University Fairfax)
- Eiji Okamoto (Tsukuba University).

From many submissions we have selected the 40 best for a presentation as full paper. The quality and quantity of submissions have improved considerably over the last years and the conference officers sometimes faced a difficult decision when selecting which papers should be accepted. This year's acceptance rate has decreased to 25% for full papers. In addition, several workshops and short papers show ongoing research projects and offer interesting starting points for discussions.

We wish all participants an enjoyable conference and interesting discussions.

General Co-chairs

Makoto Takizawa, *Sekei University, Japan*
A Min Tjoa, *Vienna University of Technology, Austria*

Message from ARES Workshops' Co-chairs

Welcome to the Workshops of the 4th International Conference on Availability, Reliability and Security (ARES) which is held at the Fukuoka Institute of Technology, Fukuoka, Japan from March 16 -19, 2009.

The workshops are very important events for ARES as they provide an essential platform for researchers of various domains to present and discuss their current results. This year we can offer the conference attendees' 10 workshops which range from "start-ups" to well-established ones supporting ARES the fourth year.

The succeeding listing comprises the workshops of ARES 2009:

1. The Forth International Workshop on Dependability Aspects on Data Warehousing and Mining applications (DAWAM-2009)
2. The Fourth International Workshop on Frontiers in Availability, Reliability and Security (FARES 2009)
3. The Third International Workshop on Secure Software Engineering (SecSE-2009)
4. The Third Workshop on Advances in Information Security (WAIS-2009)
5. The Second International Workshop on Digital Forensics (WSDF-2009)
6. The First International Workshop on Global Information Security for an Inclusive Information Society (GloSec-2009)
7. The First International Workshop on Sensor Security (IWSS-2009)
8. The First International Workshop on Organizational Security Aspects (OSA-2009)
9. The First International Workshop on Recent Innovations and Breakthroughs in Cryptography (RIBC-2009)
10. The First International Workshop on Security and Usability (SecUSAB-2009)

These workshops are organized each on specific topics and thus offer researchers the opportunity to learn from this rich multi-disciplinary experience. The Workshop Chairs would like to thank the workshop organizers for their great efforts and hard work in proposing the workshop, selecting the papers, the interesting programs and for the arrangements of the workshops during the conference days.

We are grateful to Amin Anjomshoaa for his excellent work and support with the Confdriver system. We also would like to thank the support of the webmasters' team of ARES-2009 and CISIS-2009 conferences and the local organization team at Fukuoka Institute of Technology.

We would like to give special thanks to Mr. Yoji Unoki, Chairman of Board of Trustees of FIT for hosting CISIS-2009, providing the university facilities and his continuous support. We would like to thank Fukuoka Convention Bureau for their great support, help, advices and local arrangement. We are grateful to Fukuoka City and Human Line Corporation (HLC) for the financial support. We also thank Fukuoka Institute of Technology and Secure Business Austria as sponsors of our conference.

We hope you enjoy the workshops programs and proceedings.

ARES International Conference Workshops' Co-chairs

Leonard Barolli, *Fukuoka Institute of Technology, Japan*

Stefan Jakoubi, *Secure Business Austria, Austria*

Simon Tjoa, *Secure Business Austria, Austria*

Conference Officers

General Co-chairs

Makoto Takizawa, *Sekei University, Japan*
A Min Tjoa, *Vienna University of Technology, Austria*

Program Committee Co-chairs

Arjan Durresi, *Indiana University Purdue University Indianapolis, USA*
Hiroaki Kikuchi, *Tokai University, Japan*
Edgar Weippl, *Vienna University of Technology, Austria*

Workshops Co-chairs

Leonard Barolli, *Fukuoka Institute of Technology, Japan*
Stefan Jakoubi, *Secure Business Austria, Austria*
Simon Tjoa, *Secure Business Austria, Austria*

ARES Program Committee

Jemal H. Abawajy, *Deakin University, Australia*
Rafael Accorsi, *University of Freiburg, Germany*
Andre Adelsbach, *Telindus PSF S.A., Luxembourg*
Vasilis Aggelis, *Piraeus Bank SA, Greece*
John Andrews, *Loughborough, University, United Kingdom*
Amin Anjomshoaa, *Secure Business Austria, Vienna*
Davide Balzarotti, *Eurecom - Sophia Antipolis, France*
Lisa Bartlett, *Loughborough University, United Kingdom*
Massimo Bartoletti, *Universita' di Pisa, Italy*
Bharat Bhargava, *Purdue University, USA*
Christophe Blanchet, *Centre National de la Recherche Scientifique Institut de Biologie et Chimie des Protéines, France*
Benjamin Böck, *Secure Business Austria, Vienna*
Stephane Bressan, *National University of Singapore, Singapore*
Luciano Burgazzi, *Ente per le Nuove tecnologie, l'Energia e l'Ambiente, Italy*
Kevin Butler, *Pennsylvania State University, USA*
Alexander Böhm, *University of Mannheim, Germany*
Francesco Cadini, *Polytechnic of Milan, Italy*
Lasaro Camargos, *Microsoft, USA*
Jan Camenisch, *IBM Research, Zurich*
Jiannong Cao, *Hong Kong Polytechnic University, China*
Barbara Carminati, *University of Insubria, Italy*
Jordi Castellà-Roca, *Rovira i Virgili University of Tarragona, Spain*
David Chadwick, *University of Kent, United Kingdom*
Surendar Chandra, *University of Notre Dame, USA*
Simon Christophe, *Nancy University, France*
Soon Ae Chun, *College of Staten Island/City University of New York, USA*
Nathan Clarke, *University of Plymouth, United Kingdom*
Ricardo Corin, *Microsoft Cambridge, United Kingdom*
George Davida, *University of Wisconsin at Milwaukee, USA*
Jacques Demerjian, *Communication & Systems, Homeland Security, France*
Beniamino Di Martino, *Second University of Naples, Italy*
Jochen Dinger, *Universitaet Karlsruhe, Germany*
Schahram Dustdar, *Vienna University of Technology, Austria*
Andreas Ekelhart, *Secure Business Austria, Vienna*
Christian Engelmann, *Oak Ridge National Laboratory, USA*
Yung-Chin Fang, *Dell Inc., USA*
Hannes Federrath, *University of Regensburg, Germany*
Christophe Feltus, *Centre de Recherche Public Henri Tudor, Luxembourg*
Stefan Fenz, *Secure Business Austria, Vienna*
Eduardo Fernandez Medina, *University of Castilla-La Mancha, Spain*
Vincenzo De Florio, *University of Antwerp, Belgium*
Vladimir Fomichov, *K.E. Tsiolkovsky Russian State Technological University, Russia*
Jordi Forné, *Universitat Politècnica de Catalunya, Spain*
Huirong Fu, *Oakland University, Michigan, USA*
Steven Furnell, *University of Plymouth, United Kingdom*
Javier Garcia-Villalba, *Complutense University of Madrid, Spain*
Karl Goeschka, *Vienna University of Technology, Austria*

Swapna Gokhale, *University of Connecticut, USA*
Gernot Goluch, *Secure Business Austria, Vienna*
Marcin Gorawski, *Silesian University of Technology, Poland*
Daniel Grosu, *Wayne State University, USA*
Michael Grottke, *University of Erlangen-Nuremberg, Germany*
Stephan Groß, *Technische Universität Dresden, Germany*
Le Gruenwald, *University of Oklahoma, USA*
Abdelkader Hameurlain, *Paul Sabatier University, France*
Marit Hansen, *Independent Centre for Privacy Protection, Kiel, Germany*
Yanxiang He, *Wuhan University, China*
Rattikorn Hewett, *Texas Tech University, USA*
Jimmy Huang, *York University, Canada*
Martin Gilje Jaatun, *SINTEF Information and Communication Technology, Norway*
Stefan Jakoubi, *Secure Business Austria, Austria*
Hai Jin, *Huazhong University of Science and Technology, China*
Jan Jurjens, *Munich University of Technology, Germany*
Kresimir Kasal, *Secure Business Austria, Vienna*
Stefan Katzenbeisser, *Technische Universität Darmstadt, Germany*
Holger Kenn, *University of Bremen, Germany*
Dong Seong Kim, *Duke University, USA*
Raphael Kunis, *Technische Universität Chemnitz, Germany*
Yih-Jiun Lee, *Department of Information Management, CTU, Taiwan*
Jun Li, *University of Oregon, USA*
Chae-Hoon Lim, *Sejong University, Korea*
Man Lin, St. Francis *Xavier University, Canada*
Hua Liu, *Xerox Labs, USA*
Jianhua Ma, *Hosei University, Japan*
Josef Makolm, *Federal Ministry of Finance, Austria*
Carsten Maple, *University of Luton, United Kingdom*
Keith Martin, *Royal Holloway, University of London, United Kingdom*
Rivalino Matias Jr., *Duke Electrical and Computer Engineering, USA*
Nasrullah Memon, *Aalborg University Esbjerg, Denmark*
Florian Michahelles, *ETH Zurich, Department of Management, Technology*
Geyong Min, *University of Bradford, United Kingdom*
George Mohay, *Queensland University of Technology, Australia*
Mattia Monga, *Universita` degli Studi di Milano, Italy*
Marina Mongiello, *Technical University of Bari, Italy*
Yi Mu, *University of Wollongong, Australia*
Thomas Neubauer, *Secure Business Austria, Vienna*
Jesper Buus Nielsen, *University of Aarhus, Denmark*
Thomas Nowey, *University of Regensburg, Germany*
Hong Ong, Oak, *Ridge National Laboratory, USA*
Jose A. Onieva, *Universidad de Málaga, Spain*
Maria Papadaki, *University of Plymouth, United Kingdom*
Lucia Draque Penso, *University of Mannheim, Germany*
Günther Pernul, *University of Regensburg, Germany*
Makan Pourzandi, *Ericsson Canada, Canada*
Gerald Quirchmayr, *University of Vienna, Austria*
Jean-Jacques Quisquater, *Universite Catholique de Louvain, Belgium*
Raghav Rao, *State University of New York at Buffalo, USA*
Indrajit Ray, *Colorado State University, USA*

Domenico Rosaci, *University "Mediterranea" of Reggio Calabria, Italy*
Bimal Roy, *Indian Statistical Institute, India*
Kouichi Sakurai, *Kyushu University, Japan*
Biplab Sarker, *Primal Fusion, Waterloo, Canada*
Christian Schläger, *Ernst & Young, Germany*
Rodrigo Schmidt, *École Polytechnique Fédérale de Lausanne, Switzerland*
Tony Shan, *Bank of America, USA*
Richard Sinnott, *University of Glasgow, United Kingdom*
Jill Slay, *University of South Australia, Australia*
Jon A. Solworth, *University of Illinois at Chicago*
Dieter Sommer, *IBM Research, Zurich*
Aaron Striegel, *University of Notre Dame, USA*
Tsuyoshi Takagi, *Future University, Hakodate, Japan*
Oliver Theel, *University of Oldenburg, Germany*
Marco Thorbruegge, *European Network and Information Security Agency, Greece*
Simon Tjoa, *Secure Business Austria, Vienna*
Juan-Carlos Trujillo Mondéjar, *University of Alicante, Spain*
Kalyan Vaidyanathan, *Sun Microsystems, USA*
Luca Vigano, *University of Verona, Italy*
Umberto Villano, *Universita' del Sannio, Italy*
Melanie Volkamer, *Institute of IT-Security and Security, University of Passau, Germany*
Carine Webber, *Universidade de Caxias do Sul, Brazil*
Yawen Wei, *Iowa State University, USA*
Edgar Weippl, *Secure Business Austria, Vienna*
Severin Winkler, *Secure Business Austria, Vienna*
Liudong Xing, *University of Massachusetts, USA*
Mariemma Yagüe, *University of Malaga, Spain*
Jeff Yan, *Newcastle University, United Kingdom*
Laurence T. Yang, *Saint Francis Xavier University, Canada*
Alec Yasinsac, *University of South Alabama, USA*
George Yee, *National Research Council, Canada*
Meng Yu, *Western Illinois University, Illinois*
Nicola Zannone, *University of Trento, Italy*
Jianhong Zhang, *North China University of Technology, China*
Liqiang Zhang, *Indiana University South Bend, USA*
Jianying Zhou, *Institute for Infocomm Research, Singapore*
Bo Zhu, *Concordia University, Canada*

2009 International Conference on Availability, Reliability and Security

ARES 2009

Table of Contents

Distributed Systems and Grid (ARES Full Papers)

SOA Security (ARES Full Papers)

Enterprise Security 1 (ARES Full Papers)

Intrusion and Fraud Detection (ARES Full Papers)

Enterprise Security 2 (ARES Full Papers)

Digital Forensics and Security in Communication (ARES Full Papers)

Availability and Reliability 1 (ARES Full Papers)

Cryptography (ARES Full Papers)

Software Security 1 (ARES Full Papers)

Software Security 2 (ARES Full Papers)

Availability and Reliability 2 (ARES Full Papers)

Privacy and Trust (ARES Full Papers)

Dependable Systems and Trusted Computing 1 (ARES Short Papers)

Dependable Systems and Trusted Computing 2 (ARES Short Papers)

Software Security (ARES Short Papers)

Privacy and Trust (ARES Short Papers)

Enterprise Security and Security Evaluation 1 (ARES Short Papers)

Enterprise Security and Security Evaluation 2 (ARES Short Papers)

Availability and Reliability (ARES Short Papers)

Authentication and Authorization (ARES Short Papers)

Cryptography 1 (ARES Short Papers)

Cryptography 2 (ARES Short Papers)

DAWAM 2009 - Security & Privacy Enhancement in DWHs

DAWAM 2009 - Intrusion and Network Attack Prevention

Automated Support for Security Requirements Engineering in Software Product Line Domain Engineering

Daniel Mellado
National Competition Commission.
IT & Systems Department.
Madrid, Spain.
Daniel.Mellado@uclm.es

Jesús Rodríguez, Eduardo Fernández-Medina and Mario Piattini
University of Castilla-La Mancha.
Department of IT & Systems. Alarcos Research Group.
Institute of Information Technologies & Systems.
Paseo de la Universidad, 4. 13071. Ciudad Real, Spain.
(Jesus.Rodriguez, Eduardo.FdezMedina, Mario.Piattini)@uclm.es

Abstract

Security and requirements engineering are one of the most important factor 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 all the products of a product line. However, without a CARE (Computer-Aided Requirements Engineering) tool, the application of any security requirements engineering process or methodology is much more difficult because it has to be manually performed. Therefore, in this paper, we will present a prototype of SREPPLineTool, which provides automated support to facilitate the application of the security quality requirements engineering process for software product lines, SREPPLine. SREPPLineTool simplifies the management of security requirements in product lines by providing us with a guided, systematic and intuitive way to deal with them from the early phases of product lines development, simplifying the management and the visualization of the artefacts variability and traceability links and the integration of the security standards, as well as the management of the security reference model proposed by SREPPLine. Finally we shall illustrate the application of SREPPLineTool by describing a simple example as a preliminary validation of it.

Keywords*: Security requirements, product lines, security variability, Common Criteria, security.*

1. Introduction

In the last years we have observed more and more organizations in a tight spot due to security breaches. In fact, the number of reported application vulnerabilities has risen from 171 in 1995 to 7,236 in 2007 [2], according to the statistics from the Software Engineering Institute's CERT Coordination Center. The tendency towards larger systems that are distributed over the Internet has introduced many new security threats [20], so that present-day information systems are vulnerable to a host of threats and cyber-attackers such as malicious hackers, code writers, cyber-terrorists, etc. [4].

Software product lines (SPL) have become the most successful approach in the reuse field which can help to significantly reduce time-to-market as well as development costs. In this type of software intensive systems, such as SPL, security is a cross-cutting concern and should consequently be subject to careful requirements analysis and decision making. In addition, many requirements engineering practices must be appropriately tailored to the specific demands of product lines [1].

Therefore, software security is getting more and more interesting for software engineers [24]. This has caused that the discipline of Security Requirements Engineering is highly considered as part of Security Engineering applied to the process of development of information systems that so far, has not been paid the necessary attention [14]. This discipline known as Security Requirements Engineering is a very important part of the SPL development process for the achievement of secure SPL and products, because it provides techniques, methods, standards and systematic and repeatable procedures for tackling SPL security requirement issues throughout the SPL development lifecycle both to ensure the definition of security quality requirements and to manage variability of security properties.

Nevertheless, software engineering methodologies

978-0-7695-3564-7/09 $25.00 © 2009 IEEE
DOI 10.1109/ARES.2009.23

and standard proposals of SPL engineering have traditionally ignored security requirements and security variability issues, so that there are several works that deals with security requirements management tools, similar to SREPPLineTool, although none of them are not sufficiently specific nor tailored for the SPL development paradigm, mainly because they do not deal with security requirements variability.

In this paper, we will describe the prototype of a security requirements management tool called SREPPLineTool that we have developed to provide automated support to the SREPPLine (security quality requirements engineering process for software product lines) application. SREPPLineTool will provide a guided, systematic and intuitive way for the application of SREPPLine, as well as a simple integration with the rest of requirements and the different phases of the IS development lifecycle. It also facilitates the integration of the Common Criteria (CC) [11] and ISO/IEC 27001 into the software development process as well as the fulfilment of the IEEE 830:1998 standard [8]. To do so, it is helped by using the functionalities offered by '*IBM Rational RequisitePro'* (CARE tool which is extended by SREPPLineTool). Additionally, this prototype helps to develop products and SPL which conforms to the aforementioned security standards with regard to the management of security requirements and without being necessary to perfectly know those standards and reducing the participation of security experts to get it, in other words, it improves the SREPPLine efficiency. Furthermore, thanks to the Security Reference Model implemented in SREPPLineTool, it is easier the management and the visualization of the artefacts variability and traceability links as well as the reusability of the security artifacts, thus improving quality successively.

The rest of the paper is organized as follows: In section 2, we will summarize some of the basic characteristics of SREPPLine with the aim of understanding the later explanation of the tool. Then, in section 3, we will illustrate the tool by describing a simple example of the SREPPLineTool application as a preliminary validation of it, as well we will put forward the lessons learnt. Next, in section 4, we will present the related work. Lastly, our conclusions and future work will be set out in section 5.

2. Overview of SREPPLine: security quality requirements engineering process for software product lines

A software product line is a set of software-intensive systems sharing a common, managed set of features which satisfy the specific needs of a particular market segment or mission and which are developed from a common set of core assets in a prescribed way [3]. The software product line engineering paradigm differentiates two processes: domain engineering, that is the process of SPL engineering in which commonality and variability of the product line are defined and realised; and application engineering, that 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 [21].

SREPPLine (security quality requirements engineering process for software product lines) [18] is an add-in of activities, which can be incorporated into an organization's SPL development process model providing it with a security requirements engineering approach. Because we have defined the key activities that have to be part of each SPL process. The order in which they are performed depends on the particular process that is established in an organisation. Thus, the subprocesses and their activities can be combined with existing development methods such as RUP (Rational Unified Process), or other development processes.

It is a security features or security goals based process which is driven by risk and security standards (concretely ISO/IEC 27001 [12] and Common Criteria [11]) and deals with security requirements and their related artefacts from the early stages of SPL development in a systematic and intuitive way especially tailored to SPL based development. It is based on the use of the latest and widely validated security requirements techniques, such as security use cases [5] or misuse cases [20], along with the integration of the Common Criteria (CC) components and ISO/IEC 27001 controls into the SPL lifecycle in order to facilitate SPL products security certification. Moreover, our proposed process suggests using a method to carry out the risk assessment which conforms to ISO/IEC 13335 [9], and concretely it uses MAGERIT [22] for both SPL risk assessment and SPL products risk assessment. Furthermore, SREPPLine has the aim of minimizing the necessary security standards knowledge as well as security expert participation during SPL products development. To this end, it provides a Security Core Assets Repository to facilitate security artefacts reuse and to implement the Security Reference Meta Model, which is composed of the Security Variability Sub-Meta Model and the Security Requirement Decision Sub-Meta Model, that assist in the management of the variability and traceability of the security requirements related artefacts of the SPL and its products. This meta model is the basis through which the activities of SREPPLine 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 engineering.

Our process, which is integrated into the proposed framework for SPL engineering of Pohl et al. in [21], is composed of two subprocesses with their respective activities: Product Line Security Domain Requirements Engineering (PLSecDomReq) subprocess and Product Line Security Application Requirements Engineering (PLSecAppReq) subprocess.

3. SREPPLineTool

We have developed a prototype of a CARE (Computer Aided Requirements Engineering) tool, called SREPPLineTool, which is a first approximation that will help us obtain experience of the problem through its application to real case studies to refine it and obtain a definitive version of it. SREPPLineTool prototype lets us apply the SREPPLine process in a SPL development by providing automated support to its activities. This tool implements the Security Reference Meta Model (explained in [15]) by means of dynamic repositories of security artefacts, and guides you in the execution of the process in a sequential way. Thus, it is able to propose related security artefacts at each activity of SREPPLine process depending on the domain categories of the artefacts of the SPL project. In addition, SREPPLineTool by means of wizard windows makes easier the management and the visualization of the artefacts variability and traceability links as well as the generation of the security documents of the SPL, which could be generated in XML, and the integration with other functional and non-functional requirements and features.

3.1. Developing the Tool

This prototype has been developed with .NET technology and implemented with C#, using a SQL Server 2005 database and is linked with IBM Rational RequisitePro tool by means of a Visual Basic.NET interface as it is described in Fig. 1 to retain the advantages of this requirements management tool, so that it can read the requirements and features from a RequisitePro project and it can also send the generated documents by itself to a RequistePro project.

Fig. 1 SREPPLineTool architecture

3.2. SREPPLineTool in practice

In this section, we will describe how SREPPLineTool can be applied in practice and automatizes the application of SREPPLine in a representative case of security critical SPL in which security requirements have to be correctly treated in order to achieve a robust SPL whose members (each product) would be able to manage properly private data.

This example concentrates on the results of the SREPPLineTool application to domain engineering in order to develop a SPL of a CRM (Customer Relationship Management) system, which may have several different configurations for different public institutions of the Spanish Social Security System.

Previously studied in [18] without using any security or SPL tool and under a different perspective in this case, it was carried out in the context of a reengineering process, which was performed over the SPL to adapt it to a new legal environment (a new privacy data protection legislation and new business laws of the Public Administration), so it was critical that the "new" SPL continued being secure. SREPPLineTool helped us to obtain all the security requirements artefacts of the line proposed in the sub-process PLSecDomReq of SREPPLine. This example has been simplified and summed up in order to enable points of the tool easily illustrated in this paper.

Before starting the execution of SREPPLineTool, the most important features for our SPL (PPII-CRM) had been identified and registered into IBM Rational RequistePro. Most of them were the same of the original PPII-CRM, such as: Internal Services or Citizens Services, and the most important new elicited features were the following: conformance to new data privacy protection legislation; SMS platform integration; and a new business service online to report sick leave of employees.

Next, we will describe each tab of SREPPLineTool which match up with SREPPLine activities.

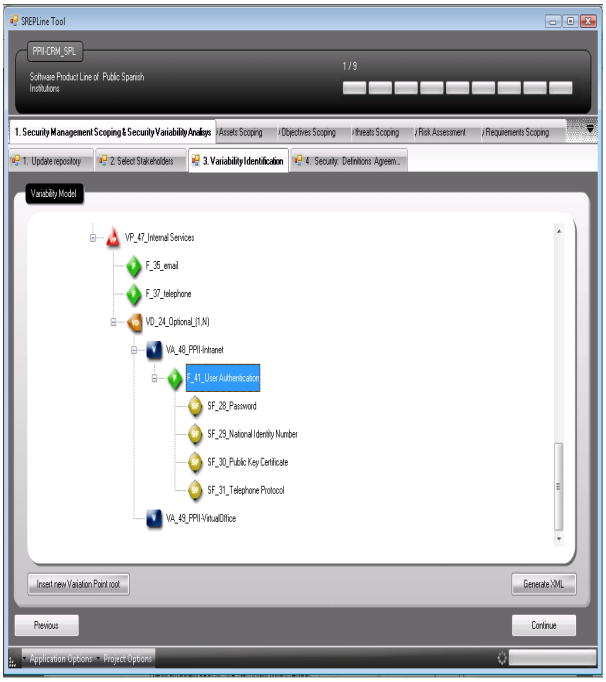

Fig. 2 Tab 1 of SREPPLineTool

Tab 1 = Activity 1.1: Security Management Scoping & Variability Analysis. The first sub-activity was the **Repository Update**, in this stage we introduced manually the previous security artefacts of the former version of the SPL (which was developed without using SREPPLineTool) and we also received a request from one of the Stakeholders asking for import the features from IBM/Rational RequisitePro. Then, the Security Requirement Engineer looked up in the Security Resources Repository of the tool in order to elicitate and to propose the security features of the product line and he suggested relating the following new security features to the feature named "data privacy protection legislation conformance".

- Strong User Authenticity
- Secure submissions of data privacy
- Confidentiality of sick leave files

Therefore, after all the suggestions had been evaluated and reviewed by the SPL manager, he linked the security features as compulsory variants of the feature "data privacy protection legislation conformance".

The next sub-activity was the **Stakeholders Selection** where the SPL manager selected the users and their roles to develop the SPL in the tool.

The third sub-activity is the **Variability Identification**, as is shown in Fig.2 SREPPLineTool allowed us to represent in a features variability tree the previously designed SPL variability model, as well as in this window of the tool we related the security artifacts obtained just before, so that we designed the security variability model throughout this interface, which can be exported to XML to be used by other tools. The last sub-activity of the first activity is the **Definitions Agreements** where SREPPLineTool helped us to reach an agreement upon a common set of security definitions such as: Information security, threat, confidentiality, etc., by providing us with the definitions of these concepts according to ISO/IEC 17799:2005 and ISO/IEC 27001. Moreover, SREPPLineTool allows us to define new standards as well as their concepts, which will be registered in the repository. It also enables us to state the evaluation assurance level (EAL) of the Common Criteria (CC), such as EAL-2 of the CC for PPII-CRM product line.

Tab 2 = Activity 1.2: Security Assets Scoping. In this activity we identified the security assets for each security feature, and the dependences between assets. For instance, for the security feature "Strong User Authenticity" we identified the following security assets, which all of them were proposed automatically by the tool except the second one because it has not been introduced in the Security Resources Repository yet: User Password; User fingerprint; National Identity Number; Social Security Number; Public Key Certificate; Telephone Protocol. Furthermore, we added a value to each asset which describes how important or critical is the asset, higher rated assets represents a more importance and a greater degradation of the product in the event in which a security breach appears in the asset. In our case study the security features related to the feature "data privacy protection legislation conformance" contains critical security information and therefore their security related assets were added as assets who represents the data information of each field (password data; value=9, identity number data; value=6, certificate data; value=9; etc.). Moreover, SREPPLineTool allowed us to define hierarchical dependencies between assets; thereby the value of the top security assets was automatically propagated throughout the tree branches by means of the security assets traceability links.

Tab 3 = Activity 1.3: Security Objectives Scoping. We selected the Security Objectives for each Asset; the tool showed us the available security objectives and the current relationships between Assets and Security Objectives, and also the value for each pair (Asset, Security Objective) that represents how important is to fulfil the Security Objective for an asset, the value scale is proposed in MAGERIT [22] (which conforms to ISO/IEC 13335 [9]) (from 0 (min) to 10 (max)). In our case study we identified the following business security objectives or security dimensions: integrity, confidentiality, availability, authenticity of service users, authenticity of data origin, accountability (or

traceability) of service use and accountability of data access. These Security Objectives will be added for each asset (User Password; User fingerprint; National Identity Number; Social Security Number; Public Key Certificate; Telephone Protocol).

Tab 4 = Activity 1.4: Security Threats Scoping. The Security Threats Scoping activity is responsible of specifying and relating the pairs of security objectives and security assets with their potential security threats that might create security breaches. The Security Resources Repository of SREPPLineTool enabled us to select the security threats from the followings sources: threats formerly introduced in the repository of the tool in this SPL or in previous SPLs, ISO/IEC 27001 Control Objectives and Common Criteria Families. When a new threat is created, we can specify a set of misuse case or/and attack-trees [20] which defines the behaviour of a threat against a pair security objective – security asset.

In our case study SREPLineTool enabled us to retrieve the threats associated with the pairs of assets and security objectives of the SPL automatically, so that we identified the following threats with their respective traceability links to their related pairs of assets and security objectives.

- Threat 1: Manipulation of configuration.
- Threat 2: Masquerading of user identity.
- Threat 3: Misuse.
- Threat 4: Re-routing of messages.
- Threat 5: Unauthorized access.
- Threat 6: Repudiation.
- Threat 7: Denial of Service.

Tab 5 = Activity 1.5: Security Risks Assessment. Once the threats were identified, we carried out the risks assessment (shown in Fig.3). In order to carry out this task, SREPPLineTool uses a technique proposed in MAGERIT [22] (techniques officially recognised by the NATO at 9th NATO cyberdefense workshop and by OECD [19]) and which is based on a quantitative analysis. First of all, and with the help of stakeholders, for each pair of asset and security objective, we estimated the likelihood of the threats (in terms of frequency of occurrence from 0 to 100), as well as the degradation of the value of an asset caused by a threat (expressed as a percentage). Finally, with these data the tool automatically calculated the impact and the risk above each pair of asset and security objective, thereby higher values indicate higher impact or risk (as is depicted in Fig 3).

Tab 6 = Activity 1.6: Security Requirements Scoping. In order to derive security requirements, each pair of asset and security objective were analysed for possible relevance together with their related threats which imply more risk, so that the suitable security requirements or the suitable package of security requirements that mitigate these threats at the necessary levels with regard to the result of the risk assessment activity. Once the relevant threats to the SPL have been selected, we elicitated those security requirements that the user believed that were necessary. To facilitate this task SREPPLineTool enabled us to:

- Select security requirements formerly introduced in the repository of the tool in this SPL or in previous SPLs or create new ones.
- Select security requirements from ISO/IEC 27001 (controls) or Common Criteria (components).
- Select or create a new requirements package / or security tests.
- Select one of the requirements packages/test and within the package/test, the desired requirements.

We selected and added to the SPL those security requirements that the security requirements engineer considered relevant to the threats previously identified. This is one of the security requirements selected from the security resources repository and which was linked to the threats 2 and 5:

SR1: Ensure User Authenticity

- The secure functions of [VP_SPL_app] shall identify and authenticate an [VP_type_user] by using [Variant] before an [VP_type_user] can bind to shell of [VP_SPL_app]. (Variant = [password | fingerprint | e-certificate])

Moreover, SREPPLineTool enables us to relate security requirements to the functional and non-functional requirements of the SPL. It also facilitates the specification of the security requirements by means of a security use case specification, with the help of parametrical templates in aspect XML to define the internal and external security variability (which is carried out at Activity 8).

Tab 7 = Activity 7: Security Requirements Negotiation and Prioritization. The aim of this activity is to automatize the security requirements prioritization according to the risk of the threats mitigated by them and the dependences between other functional and non-functional requirements. For each one of the security requirements established in the SPL, we selected which level of priority we will assign it (from 0 to 10), and then SREPPLineTool sorts the security requirements list from more to less priority.

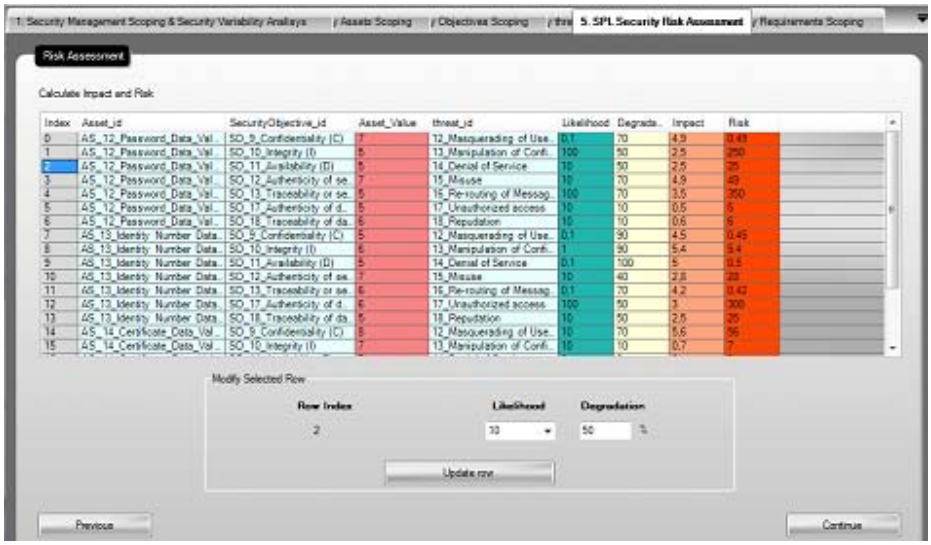

Fig. 3 Tab 8 of SREPPLineTool

Tab 8 = Activity 8: Security Requirements Specification. This activity comprised security requirements specification. In order to do so, SREPPLineTool provides us with parametrical templates in aspect XML to define the security variability of the security requirements as well as the security requirements variability links to other security requirements related (variants or variation points) and security requirements traceability links to security related artefacts.

Tab 9 = Activity 9: Security Requirements Artefacts Inspection. In this activity, SREPPLineTool facilitates the task of verifying that the security requirements conformed to IEEE 830:1998 and ISO/IEC 15408 (Common Criteria), because it made easier for the user the verification and validation of security requirements through checking those threats for which we have not specified security requirements in the SPL project, together with the assurance requirements that have not been added to the project according to the assurance level (EAL of the Common Criteria) defined in activity 1. At last, in this activity, the tool generates the SPL Protection Profile Document conforming to the Common Criteria (ISO/IEC 15408 and ISO/IEC 15446 [10]) that integrates all the information related to the rest of artifacts generated by SREPPLineTool in the previous activities. Finally, SREPPLineTool allows us to select those security artifacts modified/generated in the iteration and considered interesting for being introduced into the general repository of the tool in order to reuse these artefacts in future new SPLs.

3.3 Lessons Learnt.

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

- 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. In addition, we have to improve the graphical interface of SREPPLineTool for the security variability definition to make more intuitive this key task for security requirements engineers who are not experts in SPL engineering.
- Integration with other tools of the SPL development paradigm is essential to get an appropriate traceability of the security requirements artefacts and an appropriate implementation of the security requirements engineering into an organization.
- 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.

4. Related Work

Extensive work has been carried out on security requirements during the last few years as it was

presented in [16, 17], and there are several works that deals with security requirements management tools, similar to SREPPLineTool, although none of them are not sufficiently specific nor tailored for the SPL development paradigm, mainly because they do not deal with security requirements variability, which is an essential aspect in this paradigm. We shall now outline those tools particularly close in functionality to ours regarding security requirements.

SirenTool is an add-in of RequisitePro supporting the SIREN method [23], which is a method to elicit and specify the security system and software requirements including a repository of security requirements initially populated by using MAGERIT and which can be structured according to domains and profiles in a similar way to SREPPLine categories. Nevertheless, it only reuses requirements, which are retrieved via MAGERIT asset hierarchy or via the aforementioned repository structure. A distinguishing property of our suggestion is that we suggest using product lines, thereby by means of a Security Reference Model implemented by our tool it is reused the specifications of requirements and threats, as well as security features (typical artefacts of SPL), security objectives, assets, countermeasures and tests. In addition, the security variability can be managed in the requirements level instead of in the design level thanks to this model.

ST-Tool [7] is a CASE tool developed for modelling and analysing functional and security requirements, it allows us to design and verify them. ST-Tool has been designed to support the Secure Tropos methodology [6]. It is an agent-oriented software development tool, which manages the concepts of actor, service and social relationship. In contrast to SREPPLineTool with regards to security requirements management strictly it does not deal with security resources reuse, nor incorporate into its steps security standards integration (such as the ISO/IEC 15408 or ISO/IEC 27001) and it does not facilitate the generation of reports.

UMLsec-Tool [13] supports UMLsec. They provide an extension to the conventional process of developing use-case-oriented process for security-critical systems. They consider security aspects both in the static domain model and in the functional specification. For the elaboration of the functional aspects they introduce a question catalogue and for the domain model an UML-extension, UMLSec. However, the tool does not facilitate the explicit definition of the security variability, which is the key difference between the development of single systems and SPL engineering.

5. Conclusions and Further Work

Nowadays, software security is generating a growing interest and even more in SPL, due to the fact that security requirements issues are extremely important in SPL because a weakness in security can cause problems throughout the lifecycle of a product line. Although there have been several attempts to fill the gap between requirements engineering and SPL requirements engineering, there is not a systematic approach nor tool support available for defining security quality requirements and managing the variability of them and their related security artefacts to the models of a SPL.

While traditional requirements management tools are not able to directly support the above-exposed security requirements management in SPL engineering. We have shown in this paper that a seamless integration of security requirements engineering concepts and SPL engineering, together with the latest security requirements specification techniques (such as security use cases [5], misuse cases and attack trees [20], UMLSec [13] in the next version) and along with the most relevant security standards with regard to the management of security requirements (such as ISO/IEC 15408, ISO/IEC 27001, or ISO/IEC 17799) in these tools is possible. Thus, tools like SREPPLineTool are actually a critical enabler for the industrial uptake of security requirements engineering in SPL development, fact which was shown in the real case study performed at the Social Security of Spain [18].

Finally, there is a set of aspects planned for the future of this prototype that will allow us to increase the level of automation of SREPPLine application and so, a better efficiency of the organizations requirements engineering process in SPL engineering. Among them, we can highlight the following: to extend the type of supported requirements specifications in order to support UMLSec [13]; to refine the integration with RequisitePro and to extend the tool for it to be supported in other CARE tools; to automatize the creation of security use cases by using misuse cases created in SREPPLine PLSecDomReq activity 4; and to improve the graphical interface of SREPPLineTool for the security variability definition to make it more intuitive.

6. References

[1] A. Birk and G. Heller, "Challenges for requirements engineering and management in software product line development", *International*

Conference on Requirements Engineering (REFSQ 2007), pp. 300-305, 2007.

[2] CERT/CC, "CERT/CC Statistics 1995-2007". Pittsburgh, http://www.cert.org/stats/fullstats.html, 2008.

[3] P. Clements and L. Northrop, *Software Product Lines: Practices and Patterns*: Addison-Wesley, 2002.

[4] K.-K. R. Choo, R. G. Smith, and R. McCusker, "Future directions in technology-enabled crime: 2007–09", in *Research and Public Policy Series*, vol. 78, Australian_Government, Ed.: Australian Institute of Criminology, 2007.

[5] D. G. Firesmith, "Engineering Security Requirements", *Journal of Object Technology*, vol. 2, pp. 53-68, 2003.

[6] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone, "Requirements Engineering meets Trust Management: Model, Methodology, and Reasoning." *iTrust 2004*, pp. 176-190, 2004.

[7] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone, "ST-Tool: A CASE Tool for Security Requirements Engineering", presented at IEEE International Conference on Requirements Engineering (RE'05), 2005.

[8] IEEE, "IEEE 830: 1998 Recommended Practice for Software Requirements Specifications", 1998.

[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 27001 Information technology -- Security techniques -- Information security management systems -- Requirements." 2006.

[13] J. Jürjens, J. Schreck, and Y. Yu, "Automated Analysis of Permission-Based Security Using UMLsec", *Fundamental Approaches to Software Engineering (FASE 2008), held as part of the Joint European Conferences on Theory and Practice of Software (ETAPS 2008)*, pp. 292-295, 2008.

[14] A. v. Lamsweerde, "Elaborating Security Requirements by Construction of Intentional Anti-Models", presented at 26th International Conference on Software Engineering, Edinburgh, 2004.

[15] D. Mellado, E. Fernandez-Medina, and M. Piattini, "Security Requirements Variability for Software Product Lines", *Symposium on Requirements Engineering for Information Security (SREIS 2008) co-located with ARES 2008*, pp. 1413-1420, 2008.

[16] D. Mellado, E. Fernández-Medina, and M. Piattini, "A Comparison of the Common Criteria with Proposals of Information Systems Security Requirements", *"First International Conference on Availability, Reliability and Security" (ARES'06)*, pp. 654-661, 2006.

[17] D. Mellado, E. Fernández-Medina, and M. Piattini, "A Systematic Review of Security Requirements Engineering", in *Computers and Security (being processed)*, 2008.

[18] D. Mellado, E. Fernández-Medina, and M. Piattini, "Towards security requirements management for software product lines: a security domain requirements engineering process", in *Computer Standards & Interfaces*, vol. 30, 2008, pp. 361-371.

[19] OECD, "The promotion of a culture of security for information systems and networks in OECD countries", Organisation for Economic Co-operation and Development 2005.

[20] A. L. Opdahl and G. Sindre, "Experimental comparison of attack trees and misuse cases for security threat identification", *Information and Software Technology. In Press, Corrected Proof*, 2008.

[21] K. Pohl, G. Böckle, and F. v. d. Linden, *Software Product Line Engineering. Foundations, Principles and Techniques*. Berlin Heidelberg: Springer, 2005.

[22] Spanish_Ministry_for_Public_Administration, *Methodology for Information Systems Risk Analysis and Management*: Ministry for Public Administration, 2005.

[23] A. Toval, J. Nicolás, B. Moros, and F. García, "Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach", in *Requirements Engineering*, vol. 6, 2002, pp. 205-219.

[24] J. Viega and G. McGraw, *Building Secure Software: How to Avoid Security Problems the Right Way*. Boston: Addison-Wesley, 2002.