WORKSHOP

ERO AMERICANO

E ENGENHARIA DE REQUISITOS

E AMBIENTES DE SOFTWARE

M. Lencastre, J. F. Cunha, A. Vallecillo (Eds.)

IDEAS 2008

Proceedings of the

11th Iberoamerican Workshop on
Requirements Engineering and Software
Environments

Recife, Pernambuco, Brazil
February 11-15, 2008

Editors

Maria Lencastre
Departamento de Sistemas Computacionais
Universidade de Pernambuco
Recife, PE, Brasil
maria@dsc.upe.br

João Falcão e Cunha
Departamento de Engenharia Industrial e Gestão
Faculdade de Engenharia. Universidade do Porto
Rua Dr. Roberto Frias, s/n.
4200-465, Porto, Portugal.
jfcunha@fe.up.pt

Antonio Vallecillo
Departamento de Lenguajes y Ciencias de la Computación
Universidad de Málaga
Bulevar Luois Pasteur, 35.
29071 Málaga, Spain
av@lcc.uma.es

FICHA CATALOGRÁFICA

I12p	Iberoamerican Workshop on Requirements Engineering and Software Environments (11.: 2008 : Recife, PE) Proceedings of the 11 th Iberoamerican workshop on requirements engineering and software environments : IDEAS 2008, Recife, February 11-15, 2008. – Recife : FASA, 2008. xv, 380 p.
	1. Engenharia de software – Congressos. 2. Fórum (Debates). I. Título.
ISBN	9788570841346 CDU 004.41

Organized by Fernanda Alencar

Prefacio

El escribir un prefacio significa que se ha llegado al final de un largo camino, plagado tanto de algunos obstáculos como de gratificantes recompensas. Es para nosotros por tanto un placer darles la bienvenida a IDEAS 2008 mediante estas palabras.

El presente volumen contiene las actas con los artículos que presentado en el undécimo Workshop Americano sobre Ingeniería de Requisitos y Ambientes Software (IDEAS 2008), que se celebra este año en Recife, Pernambuco, Brasil, del 11 al 15 de Febrero de 2008.

Pernambuco ha mantenido en toda su historia una fuerte identidad, que ha contribuido decisivamente en la cultura y la política brasileña. Pernambuco fue un estado tradicionalmente centrado en la explotación de la caña de azucar. Sin embargo, en los últimos tiempos la capital de Pernambuco, Recife, se está consolidando como uno de los grandes centros tecnológicos de Brasil.

IDEAS 2008 es la undécima Conferencia de la serie IDEAS que, desde finales de los años 90 proporciona un foro para la presentación y el intercambio de resultados de la investigación y experiencias industriales en los campos de la Ingeniería de Requisitos y Ambientes de Software. En el año 2008 esta conferencia la organiza el Departamento de Sistemas Computacionales (DSC) de la Universidad Estatal de Pernambuco, junto con el Laboratorio de Ingeniería de Requisitos (LER) de la Universidad Federal de Pernambuco, en Recife, Brasil. Esta es la segunda vez que Brasil acoge a la conferencia IDEAS, tras la celebración de la primera edición en Torres, Rio Grande do Sul, en 1998.

La conferencia IDEAS trata de favorecer y promover el intercambio de conocimiento y experiencias entre profesores, estudiantes y profesionales del ámbito académico y empresarial iberoamericano, estrechando las relaciones entre los diferentes grupos de estos países que trabajan en los temas de interés de la conferencia.

Este año la conferencia recibió 74 artículos para su revisión, entre los cuales el Comité de Programa decidió seleccionar 22 para su presentación en la conferencia. Esto ha supuesto un ratio de aceptación del 29%, lo que demuestra el arduo proceso de revisión y selección al que fueron sometidos los artículos, así como la calidad de los finalmente seleccionados. Además de estos artículos, otros 12 fueron seleccionados para participar en la conferencia como artículos cortos, con la idea de favorecer y estimular el debate científico entre los asistentes y dar cabida a la presentación de trabajos incipientes. Todos los artículos fueron revisados siguiendo un sistema de peer-review por al menos 2 revisores (en media 2.84) de entre los miembros del Comité de Programa de IDEAS 2008, que estuvo compuesto por expertos internacionales de reconocido prestigio.

El programa resultante refleja perfectamente el hecho de que tanto la Ingeniería de Requisitos como los Ambientes Software involucran diferentes aspectos, tanto técnicos como de índole humana y de organización, en cuanto a recursos y a procesos. Estos aspectos incluyen los procesos de desarrollo software, los requisitos de seguridad, el uso de las ontologías en la ingeniería del software, la calidad del software, el modelado conceptual, la gestión de los requisitos, y los casos de uso y experiencias en ingeniería de software. Estos temas constituyen precisamente las sesiones del programa de la conferencia.

Por otro lado, el éxito de la conferencia IDEAS también se refleja en el número de eventos que suceden a su alrededor. IDEAS 2008 cuenta con cuatro tutoriales, dos mesas redondas, y el tercer Workshop Internacional sobre i* (istar'08). Además, este año hemos contado con tres conferenciantes invitados de primer nivel: el profesor John Mylopoulos (de las universidades de Toronto, Canada, y Trento, Italia) que impartió la charla "Goal-Oriented Requirements Engineering"; el profesor Oscar Pastor López (de la Universidad Politécnica de Valencia,España) que impartió la charla "Web Engineering: Present, Past and Future"; y el profesor José Carlos Maldonado (de la Universidad de São Paulo, Brasil) que impartió la charla "Software testing in the Context of Qualipso Project and National Perspectives". Nuestro agradecimiento más sincero por su disponibilidad para aceptar la invitación y venir a Recife a impartir sus conferencias.

También queremos expresar nuestro agradecimiento a los miembros del Comité de Programa por su tiempo y dedicación a la hora de revisar los artículos y seleccionar los artículos aceptados para su presentación, que han permitido confeccionar un año más un programa de altísima calidad y nivel. También queremos agradecerles a los organizadores locales del Departamento de Sistemas Computacionales (DSC) de la Universidad Técnica de Pernambuco todo su esfuerzo y trabajo, que han permitido hacer realidad esta conferencia. Mención especial requiere a Profa. Fernanda Alencar, que fue la encargada de confeccionar estas actas y a Prof. Jaelson Castro por su apoyo constante y ayuda.

Finalmente, nos gustaría mencionar nuestro agradecimiento explícito a los patrocinadores del evento: El Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), la Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (FACEPE), la Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), la Pro-reitoria para Assuntos de Pesquisa e Pós-Graduação (Propesq-UFPE), y el Departamento de Sistemas Computacionais (DSC/POLI/UPE) que hicieron posible que la conferencia fuera todo un éxito. También mencionar el sistema de revisión de artículos que utilizamos, EasyChair, que fue de una utilidad y ayuda inestimable. Nos gustaría por tanto agradecer a su creador, Andrei Voronkov, por toda su ayuda y eficiente soporte durante el proceso de revisión y la preparación de las actas.

Muchas gracias a todos los asistentes y participantes a IDEAS 2008, y esperamos verles de nuevo en Colombia en el próximo IDEAS 2009.

Diciembre 2007

Maria Lencastre
João Falcão e Cunha
Antonio Vallecillo

Conference Organization

General Chair

Maria Lencastre

Programme Chairs

João Falcão e Cunha
Antonio Vallecillo

Programme Committee

Fernanda Alencar
João Paulo Almeida
Carina Alves
João Araújo
Alvaro Arenas
Marcio Barros
Nelly Bencomo
Pere Botella
Regina Braga
Antonio Brogi
Coral Calero
Rafael Calvo
Carlos Canal
Jaelson Castro
Alejandra Cechich
Luca Cernuzzi
Marcio Delamaro
Isabel Diaz
Amador Duran
Sandra Fabbri
Ricardo Falbo
Xavier Franch
Marcelo Frias
Alessandro Garcia
Jesús García Molina
Itana Maria de Souza Gimenes
Silvia Gordillo
Juan Hernandez
Miguel Katrib
Nora Koch
Julio Cesar Leite
Maria Lencastre
José Carlos Maldonado
Esperanza Marcos
Emilia Mendes
Jonas Montilva
Ana Moreira
Nuno Nunes
Hanna Oktaba
Luis Olsina
Oscar Pastor López
Vicente Pelechano
Ernesto Pimentel
Francisco Pinheiro
Claudia Pons
Ruben Prieto-Díaz
Daniel Riesco
Gustavo Rossi
Francisco Ruíz
Victor Santander
Ernest Teniente
Miguel Toro Bonilla
Guilherme Travassos
Alexandre Vasconcelos
Marcello Visconti

Local Organization

Alex Sandro Gomes
Carina Alves
Cristine Gusmão
Fernanda Alencar

Genésio Neto
Jaelson Castro
Luis Soares
Márcia Lucena
Márcio Cornélio
Ricardo Massa
Ricardo Ramos
Tiago Massoni
Sérgio Soares

External Reviewers

Marcio Barros
Regina Braga
Guillermo Juan Covella
Maria Istela Cagnin
Rafael Calvo
Valter Vieira de Camargo
Pedro J. Clemente
Jose Maria Conejero
Javier Cubo
Javier Cámara
Leandro Da°on
Amador Duran
Isabel Diaz
Maria Jose Escalona
Sandra Fabbri
Sandra Ferrari
Andrés Flores
Fred Freitas
Thaizel Fuentes
Roxana Giandini
Itana Gimenes
itana Gimenes
Silvia Gordillo
Francisco Gutiérrez
Francisco Hernández-Quiroz
Maria de los Angeles Martin
Hernan Melgratti
Hernan Molina
German Montejano
Ana Moreira
Hanna Oktaba
J.L. Ortega-Arjona
Joaquin Peña
Antonia M. Reina Quintero
Fernando Rincón
Gustavo Rossi
Gwen Salaui
Marisol Sanchez-Alonso
Laura Semini
Flavio Signorelli Mendes
Simone do Rocio Senger de Souza
Rosana Teresinha Vaccare Braga
Jo Ueyama
Rafael Valencia
David Benavides
Valeria de Castro

Table of Contents

Short papers

Aplicando un Proceso de Ingeniería de Requisitos de Seguridad de Dominio para Líneas de Producto Software

Daniel Mellado[1], Eduardo Fernández-Medina[2] y Mario Piattini[2]

[1] Ministerio de Trabajo y Asuntos Sociales; Gerencia de Informática de la Seguridad Social; Centro de Desarrollo del Instituto Nacional de la Seguridad Social; Madrid, España.
Daniel.Mellado@alu.uclm.es

[2] Grupo Alarcos, Dpto. de Tecnologías y Sistemas de Información, Centro Mixto de Investigación y Desarrollo de Software UCLM-Indra; Universidad de Castilla-La Mancha.
Paseo de la Universidad 4, 13071 Ciudad Real, España.
(Eduardo.Fdez-Medina, Mario.Piattini)@uclm.es

Resumen. La gestión de los requisitos de seguridad es especialmente importante en las líneas de producto software, debido a que una brecha o vulnerabilidad de seguridad puede provocar problemas a todos los productos de la línea y afectar a todo el ciclo de vida. La principal contribución de este trabajo es ilustrar a través de un escenario de aplicación real, cómo de una forma guiada, sistemática e intuitiva se pueden tratar los requisitos de seguridad y facilitar su gestión desde las primeras fases del desarrollo basado en líneas de producto software, mediante la aplicación de SREPPLine. El cual es un proceso de ingeniería de requisitos de seguridad que hemos desarrollado y que está particularizado para el desarrollo de líneas de producto software seguras con el fin de facilitar la compleja gestión de la variabilidad y reutilización, así como las relaciones de trazabilidad de los requisitos de seguridad en éstas. Para lo cual se propone la utilización de las últimas técnicas de variabilidad de requisitos en líneas así como las técnicas de requisitos de seguridad, junto con la integración de los Criterios Comunes (ISO/IEC 15408). De esta forma se facilita que la línea y sus productos sean conformes con los estándares de seguridad más relevantes (ISO/IEC 27001, ISO/IEC 17799, ISO/IEC 15408 e ISO/IEC 21827) en lo relativo a la gestión de requisitos de seguridad.

Palabras clave: Ingeniería de Requisitos; Requisitos de Seguridad; Líneas de Producto Software; Criterios Comunes; ISO 27001.

1 Introducción

En la actualidad, está ampliamente defendido el principio que establece que la seguridad debería considerarse desde las primeras fases del desarrollo y que los requisitos de seguridad deberían definirse junto con los demás requisitos del SI, como se recoge en varios estudios recientes [11, 14, 19],ya que esto permite soluciones más eficientes y robustas así como ayuda a reducir los conflictos entre los requisitos de seguridad y los demás requisitos. Asimismo, en los últimos años se está observando

un incremento en la demanda de software y en su complejidad requerida, lo cual aumenta la potencialidad de presentar brechas de seguridad [25]. Por esto, hoy, para poder alcanzar los niveles deseados de calidad y mejorar la productividad multitud de sistemas se están desarrollando basándose en el paradigma de ingeniería de Líneas de Producto Software (LPS), ya que las LPS ayudan a reducir significativamente el tiempo de puesta en producción y los costes de desarrollo, mediante la reutilización de todo tipo de artefactos [3, 4].

Debido a la complejidad y a la naturaleza extensiva de las LPS, la seguridad y la ingeniería de requisitos son mucho más importantes para la puesta en práctica del desarrollo basado en LPS, de lo que ya son para el desarrollo de un Sistema de Información (SI), ya que una brecha de seguridad o vulnerabilidad en la línea puede provocar importantes problemas a largo plazo a todos los productos de la misma [9]. Es por ello que la disciplina conocida como Ingeniería de Requisitos de Seguridad [12], sea una parte muy importante en el proceso de desarrollo software y especialmente dada su complejidad para conseguir LPS seguras, ya que facilitan técnicas, métodos y normas para abordar esta tarea desde las primeras fases del desarrollo e implica el uso de procedimientos repetibles y sistemáticos para asegurar que el conjunto de requisitos obtenidos es completo, consistente y fácilmente comprensible y analizable por parte de los diferentes actores implicados en el desarrollo de la LPS y sus sistemas.

Después de analizar en [17, 18] las propuestas más recientes y relevantes relativas a los requisitos de seguridad en SI como: [6, 7, 15, 21, 23, 24], etc.; junto con las propuestas más importantes sobre gestión de requisitos en LPS, como [4, 9, 10, 20, 22], así como las arquitecturas de seguridad de referencia para LPS, como [1, 5, 8], llegamos a la conclusión de que las propuestas existentes no eran lo suficientemente específicas para facilitar la gestión de requisitos de seguridad en LPS de una forma sistemática e intuitiva y estaban orientadas a la solución en lugar de a la ingeniería de requisitos de seguridad. Por ello, desarrollamos el proceso de ingeniería de requisitos de seguridad para LPS, SREPPLine (Security Requirements Engineering Process for software Product Lines) [16], cuyo objetivo es facilitar una integración concreta de las actividades relativas a la gestión de requisitos de seguridad en el resto de actividades del desarrollo basado en LPS y proporcionar un soporte metodológico específico para la gestión de requisitos de seguridad y del modelo de variabilidad de seguridad de la línea. Asimismo, ayuda a que las LPS y los sistemas que de ella se deriven sean conformes respecto a la gestión de requisitos de seguridad con los estándares de seguridad internacionales más importantes (como ISO/IEC 15408, ISO/IEC 17799, ISO/IEC 27001 o ISO/IEC 21827).

Por último y dada la actual escasez de literatura que describa casos de estudio reales donde se describa la gestión de los requisitos de seguridad en LPS, en este artículo se presenta un escenario real de aplicación SREPPLine [16], a fin de realizar una validación preeliminar de la aplicabilidad del mismo y verificar cómo nuestro proceso facilita la actual gestión de requisitos de seguridad en LPS y sus actividades correspondientes.

El resto del artículo está organizado de la siguiente forma: en la sección 2, se resumen los principales conceptos de la ingeniería de requisitos en las LPS. A continuación, en la sección 3, se describe de forma general el proceso SREPPLine. Segui-

damente en la sección 4, se describirá la aplicación del proceso SREPPLine en un escenario real. Y por último, en la sección 5, presentamos nuestras conclusiones y trabajos futuros.

2 La Ingeniería de Requisitos en las Líneas de Producto Software

Una Línea de Producto Software es "un conjunto intensivo de sistemas software que comparten un conjunto común y gestionado de características (features, entendidas como una característica visible para el usuario final del sistema), dónde estas características están pensadas para satisfacer las necesidades especificas de una misión o de un segmento de mercado. Asimismo, los productos son desarrollados de una forma pre-establecida a partir de un conjunto común de componentes" [4].

El paradigma de ingeniería de Líneas de Producto Software se compone de dos procesos: ingeniería del dominio e ingeniería de la aplicación [20]. La ingeniería del dominio es el proceso de ingeniería de Líneas de Producto Software en el que se define la variabilidad y elementos comunes. La ingeniería de aplicación es el proceso de ingeniería de Líneas de Producto Software en el que se construyen las aplicaciones de la línea reutilizando los artefactos del dominio y aprovechándose de la variabilidad de la Línea de Producto Software.

Por lo tanto, los requisitos de una línea de productos definen los productos de dicha línea y sus características comunes y variables [4]. La gestión de requisitos para líneas de productos debe gestionar los requisitos de la línea de productos y los requisitos de los productos concretos de la línea. Se tiene que hablar por tanto de gestión de requisitos del dominio y gestión de requisitos de la aplicación, siguiendo a [20]. La gestión de requisitos para líneas de productos debe incorporar un mecanismo mediante el cual el conjunto de requisitos para un producto concreto sea producido de manera fácil y rápida a partir de los requisitos de la línea de productos.

3. Descripción General de SREPPLine: Proceso de Ingeniería de Requisitos de Seguridad para Líneas de Producto Software

El Proceso de Ingeniería de Requisitos de Seguridad para Líneas de Producto Software (SREPPLine) [16] es un add-in de actividades (que se descomponen en tareas, donde se generan artefactos de entrada y salida, y con la participación de distintos roles) que se integran sobre el proceso de desarrollo de LPS existente en una organización, proporcionándole un enfoque en ingeniería de requisitos de seguridad específico para LPS. Los sub-procesos y actividades descritos en este artículo se pueden combinar con los procesos de desarrollo como el Proceso Unificado u otros. En este artículo describiremos la integración de SREPPLine en el marco de trabajo de ingeniería de LPS propuesto por Pohl et al. en [20].

SREPPLine es un proceso basado en activos y dirigido por el riesgo para el establecimiento de requisitos de seguridad en el desarrollo de LPS seguras. Básicamente este proceso facilita la integración los Criterios Comunes (CC) y los controles de la ISO/IEC 27001 en el desarrollo de LPS junto con el uso de un repositorio de recursos de seguridad para facilitar la variabilidad y reutilización de requisitos, activos, amenazas, test y contramedidas en la LPS. Asimismo, facilita la gestión del modelo de variabilidad de la seguridad y los distintos tipos de trazabilidad implicados entre los artefactos de seguridad entre sí, así como entre los de las aplicaciones con los de la línea. Igualmente ayuda a que la LPS y las aplicaciones o sistemas de información de dicha LPS sean conformes a los estándares de seguridad actualmente más relevantes relativos a la gestión de requisitos de seguridad (como ISO/IEC 15408, ISO/IEC 27001, ISO/IEC 17799 o ISO/IEC 21827) y tratando de minimizar la participación de los expertos de seguridad en el desarrollo de los productos y el conocimiento de los estándares.

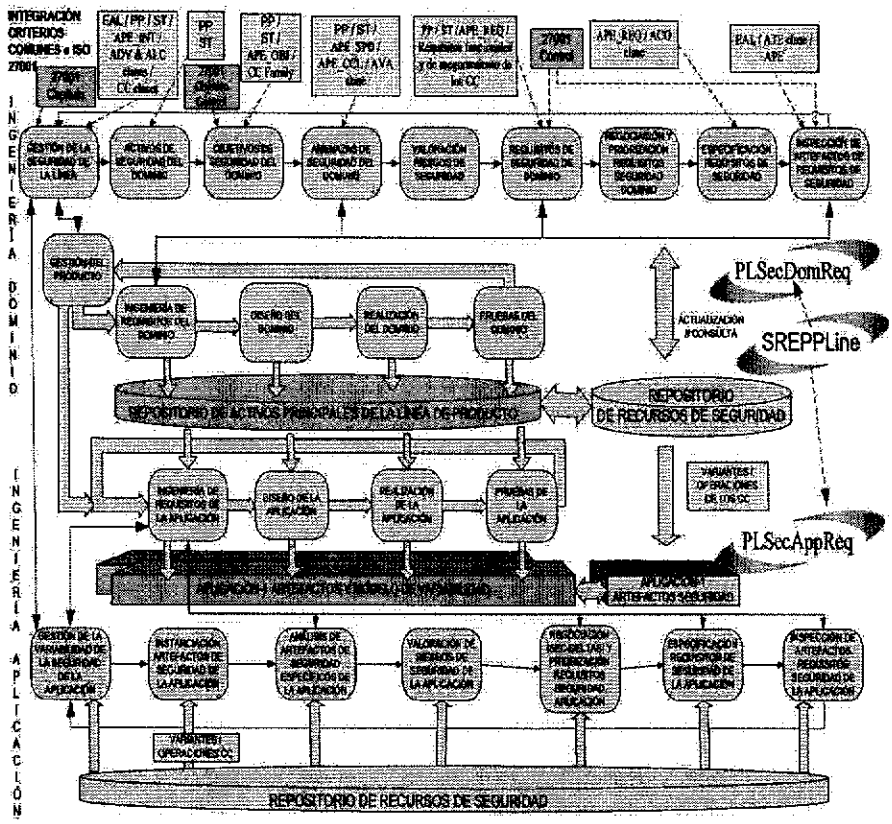


Fig. 1 Marco de Trabajo para la Ingeniería de Requisitos de Seguridad en Líneas de Producto

Como se puede observar en la Fig. 1, SREPPLine se compone de dos subprocesos con sus respectivas actividades: PLSecDomReq (Product Line Security Domain Requirements Engineering sub-process) y PLSecAppReq (Product Line

Security Application Requirements Engineering sub-process). Estos sub-procesos cubren las cuatro fases básicas de la ingeniería de requisitos [12]: elicitación de requisitos; análisis y negociación de requisitos; documentación de requisitos; y validación y verificación de requisitos. Estos sub-procesos se ejecutarán al menos por cada iteración del proceso de ingeniería del dominio y/o de la aplicación de la LPS, respectivamente. Sin embargo, dadas las restricciones de espacio, en este artículo se describirán de forma general y sin iteraciones como se aplican en la práctica las actividades del sub-proceso PLSecDomReq en un escenario real con el objetivo de facilitar una comprensión clara y global de la aplicación de dicho proceso.

Asimismo, como se observa en la Fig. 1, el Repositorio de Recursos de Seguridad se debe de integrar en el repositorio de activos comunes de la LPS, para posibilitar las relaciones de trazabilidad entre el modelo de variabilidad de la LPS y los diferentes tipos de artefactos de seguridad y otros artefactos de desarrollo, así como la trazabalidad entre los artefactos de la línea y los productos. El modelo de variabilidad de seguridad implementado por SREPPLine se apoya en el concepto de modelo de variabilidad ortogonal [20], lo cual nos permite flexibilidad para aplicarlo, ya que permite que el proceso se integre con otros modelos de desarrollo software (como modelos de 'features', de casos de uso, de diseño, o modelos de componentes o de pruebas).

4. Aplicando SREPPLine en la Práctica

En esta sección se describe cómo el subproceso de SREPPLine, PLSecDomReq, puede aplicarse en la práctica en un escenario real.

4.1 Escenario de aplicación

Se utilizará nuestro proceso propuesto (PLSecDomReq) para obtener una especificación de los requisitos de seguridad junto con sus artefactos relacionados de una línea de producto de sistemas CRM (Customer Relationship Management) para la Seguridad Social española, cuya arquitectura se muestra en la Fig. 2. Dichos sistemas deberán tener configuraciones diferentes para cubrir las particularidades de tres instituciones públicas del sistema de seguridad social de España. Por tanto, el sistema, al que llamaremos SS-CRM, se desarrollará orientado a la creación de una LPS cuyos miembros/productos variarán por configuración para adaptarse a las necesidades particulares de cada institución aunque conservando un núcleo de funcionalidades comunes y una serie de servicios específicos propios de cada CRM de cada institución. Obviamente, dado el limitado alcance de la variabilidad de la LPS se trata de un caso a modo representativo, que sirve como ejercicio instructivo de aplicación de nuestro proceso en un escenario real, teniendo en cuenta el hecho de que este caso de estudio se ha simplificado y resumido para permitir ajustarse a las restricciones de espacio y así ilustrar fácilmente los puntos principales del subproceso PLSecDomReq en este artículo.

El departamento de tecnologías de información del Organismo será el responsable del desarrollo de la LPS denominada SS-CRM así como del desarrollo de los

sistemas CRM derivados de la misma. Previamente, se cargará en el Repositorio de Recursos de Seguridad un perfil básico de seguridad de LPS genérico para el Organismo, con los artefactos de seguridad más habituales en los sistemas actuales del mismo, como los requisitos legales, normas internas, política de seguridad de la organización así como los requisitos de seguridad de los Criterios Comunes (CC) y los controles de la ISO/IEC 27001 tal y como describe SREPPLine para su aplicación.

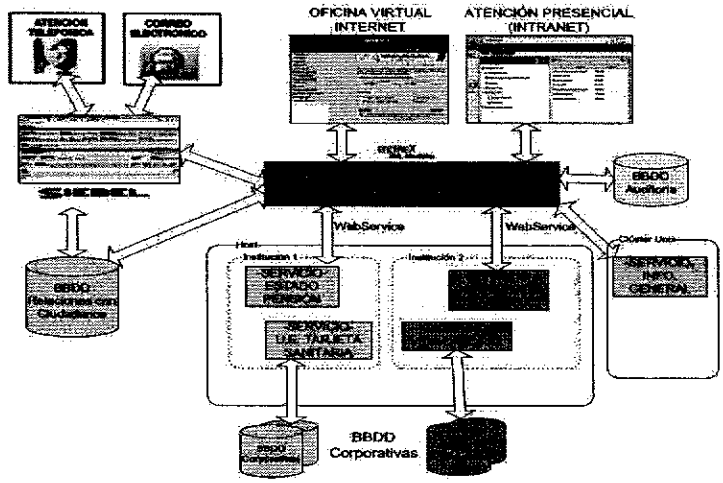


Fig. 2 Esquema de la arquitectura de la LPS de sistemas CRM en Seg-Social de España

4.2 Aplicación de SREPPLine

SP1 - Actividad A1.1: Gestión de las Características de Seguridad de la Línea. Como entrada de esta actividad se recibió el modelo de variabilidad en un árbol de características del SS-CRM en el que se describía la variabilidad de los componentes funcionales del dominio. A partir de este modelo de características identificamos las características de seguridad y sus dependencias y se desarrollo el modelo ortogonal de variabilidad de la seguridad, el cual se especificó usando XML. También se seleccionaron las clases de los CC y capítulos de la norma ISO 27001 relacionadas a dichas características de seguridad. En la Fig. 3 se muestra parte del modelo de variabilidad de seguridad, en el que se representa las características básicas de la LPS SS-CRM junto con sus activos relacionados así como la característica de seguridad de autenticación de usuario y sus relaciones con los diferentes activos según las variantes.

Se identificaron los siguientes objetivos o dimensiones de seguridad [13]: integridad (I), confidencialidad (C), disponibilidad (D), autenticidad del usuario del servicio (A_S), autenticidad del origen de los datos (A_D), trazabilidad del uso del servicio (T_S) y trazabilidad del acceso a los datos (T_D). Además, después de analizar la política de seguridad de la organización, sus procesos de negocio, los casos de uso de negocio del SS-CRM y entorno de la LPS, se identificaron los siguientes tipos o categorías de características que engloban a los activos, como se muestra en la prime-

ra columna de la Tabla 1: Servicios finales de negocio y datos de negocio, servicios internos y equipamiento (hardware, software y comunicaciones). Aunque no se tendrá en cuenta esta última categoría a fin de simplificar la aplicación y comprensión de SREPPLine.

También se llegó al acuerdo sobre varias definiciones de conceptos de seguridad así como el nivel de seguridad requerido para la LPS en base a los CC (estableciéndose el nivel de conformidad EAL 2 de los CC como mínimo para la LPS y por tanto para los sistemas que se deriven de la misma), al igual que se decidió dado el tipo de información que gestionarán los sistemas SS-CRM que se tendrá que cumplir con la legislación española de protección de de datos de carácter personal en lo referente a datos clasificados con nivel medio y alto.

SP1 - Actividad A1.2: Activos de Seguridad del Dominio de la LPS. En esta actividad se identificaron los activos de seguridad comunes y variables para cada una de las características de seguridad identificadas anteriormente, así como se establecen las dependencias entre ellos. En la primera columna de la Tabla 1 se listan parte de los activos de seguridad (identificados con '(A)' delante), categorizados por característica de seguridad. Además, en la Fig. 3 se representa en el modelo de variabilidad de seguridad sus dependencias. Asimismo, nos ayudamos del Repositorio para la realización de la identificación y categorización de los activos de seguridad, de manera que si la característica de seguridad identificada en la actividad anterior estaba ya previamente en el repositorio, éste te proponía los activos de seguridad relacionados con dicha característica. En las siguientes actividades el repositorio de recursos de seguridad se puede usar de la misma manera para la identificación del resto de artefactos de seguridad: objetivos de seguridad, amenazas y requisitos.

SP1 - Actividad A1.3: Objetivos de Seguridad del Dominio de la LPS. En esta actividad a partir de las dimensiones de seguridad identificadas en la actividad A1.1, se establecieron los objetivos de seguridad para cada uno de los activos de seguridad (junto con el análisis de variabilidad y elementos comunes, y a la vez que se determinaron los objetivos de control de la ISO 27001 y familias de los CC) así como se valoró cualitativamente cada activo de seguridad con cada uno de sus objetivos de seguridad relacionados. Para la realización de esta tarea, se realizaron entrevistas con los distintos interesados utilizando el método de evaluación Delphi y la escala de valoración propuesta en MAGERIT [13] (de 0 a 10), ya que iba a ser el método de análisis y gestión de riesgos que se utilizara a continuación. De esta forma y siguiendo el modelo cualitativo de MAGERIT únicamente los activos más altos en el árbol de dependencias entre activos obtenido en la actividad anterior son evaluados explícitamente, de manera que esta valoración se propaga automáticamente (matemáticamente) a través del árbol de dependencias del modelo de variabilidad de seguridad. Con lo que se obtuvo como resultado una tabla con los valores acumulados para cada uno de los activos identificados en la LPS. Parte de esta tabla de valores acumulados de los activos se muestra en Tabla 1. En dicha tabla, el primer valor de cada celda se corresponde con la valoración del activo y si dicho número se encuentra entre corchetes indica que se trata de una valoración propagada. Por último, se registró formalmente en XML los objetivos de seguridad y su valoración para cada uno de los activos.

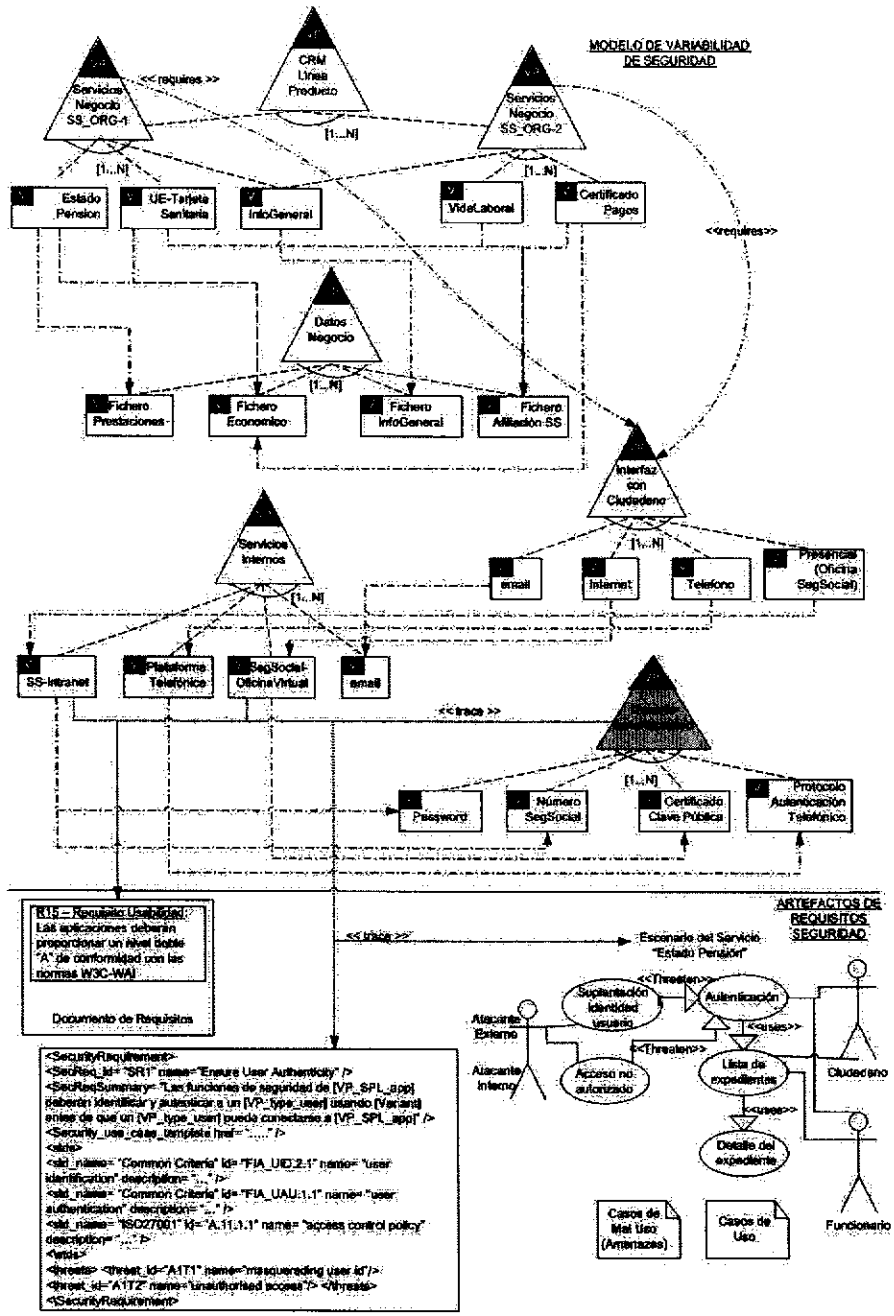


Fig. 3 Ejemplo: Modelo de variabilidad de seguridad y artefactos de los requisitos de seguridad

SP1 - Actividad A1.4: Amenazas de Seguridad del Dominio de la LPS. Como entrada de esta actividad se recibió la lista de las vulnerabilidades, amenazas y patrones de ataque más comunes en la Organización así como el catalogo de amenazas listado en la Organización. Con todos estos datos, junto con la ayuda del repositorio (que te propone posibles amenazas dados los activos que se tienen identificados) y después de analizar los casos de uso de negocio, se desarrollaron los casos de mal uso y se identificaron a la vez los atacantes potenciales con la participación del experto de seguridad. Seguidamente, se identificaron las amenazas comunes para cada activo de seguridad y utilizándose para su especificación las plantillas de casos de mal uso, y por último se incluyeron en el modelo de variabilidad de seguridad, estableciéndose las relaciones de trazabilidad correspondientes con los objetivos y activos de seguridad.

Algunas de estas amenazas se listan en la Tabla 1, donde se listan las amenazas intencionadas que amenazan al activo "Servicio de Negocio de Estado Pensión". Además, en la Fig. 3 se muestra en un ejemplo de un escenario con un caso de mal uso donde se muestran parte de los casos de mal uso del escenario del punto de variación "Autenticación" para el caso de la variante "Servicio de Negocio de Estado Pensión".

SP1 - Actividad A1.5: Valoración de Riesgos de Seguridad de la LPS. En esta actividad se realizó la estimación de riesgos utilizando MAGERIT [13]. Por lo tanto, en primer lugar se estimó la probabilidad de ocurrencia de las amenazas para cada uno de los activos relacionados (en términos de frecuencia de ocurrencia de 0 a 100: 100 para muy frecuentes, diarios; 10 para frecuentes, mensuales; 1 para normal, anual; 0'1 para poco frecuente, cada varios años). Así como se estimó el grado de degradación del activo sobre su valor expresado en porcentaje en caso de que se materializara la amenaza, para lo cual se contó con la ayuda del repositorio y con la información histórica de la Organización. A continuación, el impacto acumulado de los activos se estimó teniendo en cuenta el valor acumulado de los activos y el grado de degradación que causarían las amenazas. Con lo que seguidamente, el riesgo acumulado para los activos se estimó considerando tanto el impacto acumulado como la frecuencia de ocurrencia de cada amenaza. En la Tabla 1 se muestra parte de la estimación de impactos acumulados como parte del análisis de riesgos realizado (clasificándose el riesgo en un rango desde: 0, casi nulo; 1-2 para riesgo bajo; 3 para riesgo medio; 4 para riesgo alto; y 5 para riesgo muy alto). En dicha tabla el segundo número que aparece en cada una de las celdas se refiere al factor de degradación en los activos que causaría la amenaza correspondiente, el tercer valor se refiere al impacto acumulado en el activo y el último (cuarto) valor hace referencia al riesgo acumulado sobre el activo.

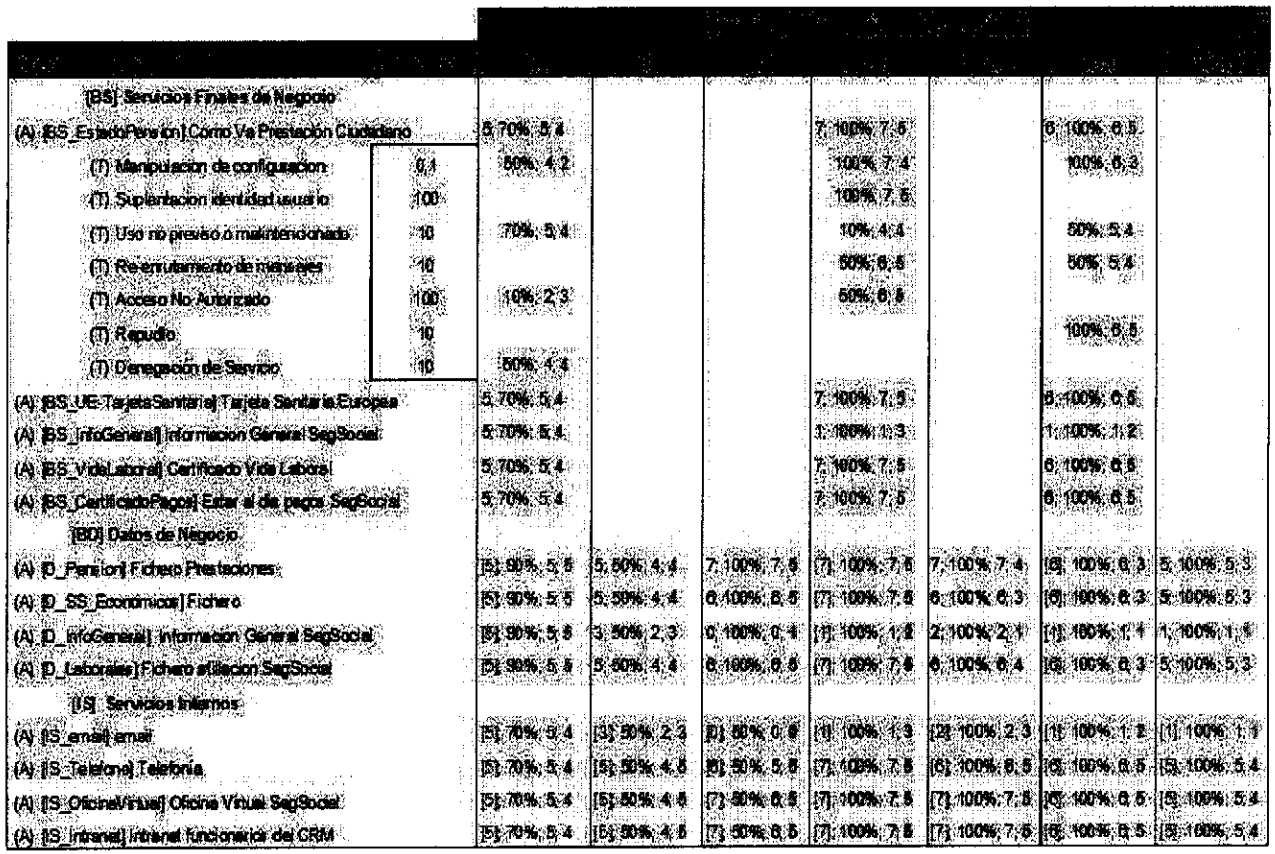
SP1 - Actividad A1.6: Requisitos de Seguridad de Dominio de la LPS. En esta actividad, como primer paso se analizaron los casos de mal uso y lo que suponía las amenazas relacionadas. Seguidamente y con la ayuda del repositorio se seleccionaron los requisitos de seguridad funcionales de los Criterios Comunes y los controles de seguridad de la ISO/IEC 27001 adecuados para mitigar las amenazas de la línea de producto software. Después, realizamos la identificación de los requisitos de seguridad comunes según los requisitos elicitados y con el análisis de riesgos realizado

anteriormente, se determinaron los requisitos de seguridad variables y se definió su variabilidad interna así como sus relaciones de dependencia, al mismo tiempo que se establecen las operaciones permitidas sobre los requisitos funcionales de seguridad por los sistemas que se deriven de la LPS, en el caso de los CC las operaciones serán: iteración, asignación, selección o refinamiento. Finalmente se modelaron los requisitos de seguridad usando casos de uso de seguridad y se establecieron las relaciones de trazabilidad correspondientes entre ellos y sus artefactos asociados (test de seguridad y medida/métrica de seguridad, amenaza, etc.) según el modelo de variabilidad definido en SREPPLine [16]. En la Fig. 3 se muestra un ejemplo de un requisito de seguridad que usando la plantilla para casos de uso de seguridad y especificándose con XML para el escenario con la variante "Servicio de Negocio de Estado Pensión".

SP1 - Actividad A1.7: Priorización y Negociación de los Requisitos de Seguridad del Dominio de la LPS. En esta actividad priorizamos los requisitos en función al riesgo estimado de las amenazas relacionadas. Seguidamente, se identificaron y se especificaron en nuestro modelo de variabilidad de seguridad las interdependencias de los requisitos de seguridad con otros requisitos funcionales y no funcionales mediante el análisis de los casos de uso y del modelo de características (en la Fig. 3 se muestra un ejemplo de interdependencia entre un requisitos de seguridad y otro tipo de requisito). Además realizamos un somero análisis coste-beneficio valorando por un lado el coste que supondría implementar cada una de las contramedidas asociadas a los requisitos de seguridad y el riesgo que supondría su no implementación. De manera que llegamos al acuerdo con los interesados en que se tendrían en cuenta aquellos requisitos de seguridad que tuvieran asociados amenazas que supongan un riesgo calificado como alto o muy alto, sea cual sea el conflicto con otros requisitos o su coste (dentro de lo que se determinó como razonable o estratégico). Sin embargo, para los requisitos de seguridad con menos riesgo se tuvieron que llegar a acuerdos cuando entraban en conflicto con otros requisitos (como por ejemplo como se muestra en la Fig. 3, el requisito de usabilidad).

SP1 - Actividad A1.8: Especificación de Requisitos de Seguridad de Dominio de la LPS. Durante esta actividad se modelaron y especificaron los requisitos de seguridad formalmente. Para ello se uso la técnica de los casos de uso de seguridad y plantillas XML parametrizadas para permitir la variabilidad y los enlaces de trazabilidad requeridos por el modelo de variabilidad de seguridad. En la Fig. 3 se puede ver un ejemplo de parte de la especificación de requisitos de seguridad especificados usando la técnica de especificación requisitos en XML orientada a aspectos así como las correspondientes trazas al modelo de variabilidad de seguridad y a la plantilla del caso de uso de seguridad utilizado.

SP1 - Actividad A1.9: Inspección de Artefactos de Requisitos de Seguridad de Dominio de la LPS. En esta actividad verificamos el grado de conformidad de la LPS con los controles de la norma ISO/IEC 27001 y a los requisitos de aseguramiento de los CC (ISO/IEC 15408) correspondientes al EAL2, así como se valida que los requisitos sean conformes al estándar IEEE 830-1998. Además, se estima el riesgo residual de la LPS para evaluar la eficacia de los requisitos de seguridad y sus contramedidas asociadas (siguiéndose de esta manera el modelo Plan-Do-Check-Act).



[BS] Servicios Finales de Negocio								
(A) [BS_EstadoPension] Como Va Prestación Ciudadano		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
(T) Manipulacion de configuracion	0,1	50%; 4; 2			100%; 7; 4		100%; 6; 3	
(T) Suplantacion identidad usuario	100				100%; 7; 5			
(T) Uso no previsto o malintencionado	10	70%; 5; 4			10%; 4; 4		50%; 5; 4	
(T) Re-enrutamiento de mensajes	10				50%; 6; 5		50%; 5; 4	
(T) Acceso No Autorizado	100	10%; 2; 3			50%; 6; 5			
(T) Repudio	10						100%; 6; 5	
(T) Denegación de Servicio	10	50%; 4; 4						
(A) [BS_UE-TarjetaSanitaria] Tarjeta Sanitaria Europea		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
(A) [BS_InfoGeneral] Informacion General SegSocial		5; 70%; 5; 4			1; 100%; 1; 3		1; 100%; 1; 2	
(A) [BS_VidaLaboral] Certificado Vida Laboral		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
(A) [BS_CertificadoPagos] Estar al dia pagos SegSocial		5; 70%; 5; 4			7; 100%; 7; 5		6; 100%; 6; 5	
[BD] Datos de Negocio								
(A) [D_Pension] Fichero Prestaciones		[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_Economicos] Fichero		[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_InfoGeneral] Informacion General SegSocial		[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_Laborales] Fichero afiliacion SegSocial		[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
[S] Servicios Internos								
(A) [S_email] email		[5]; 70%; 5; 4	[3]; 50%; 2; 3	[0]; 50%; 0; 5	[1]; 100%; 1; 3	[2]; 100%; 2; 3	[1]; 100%; 1; 2	[1]; 100%; 1; 1
(A) [S_Telefonia] Telefonia		[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) [S_OficinaVirtual] Oficina Virtual SegSocial		[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) [S_Intranet] Intranet funcionarios del CRM		[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

Tabla 1 Parte del mapa de análisis de riesgos de SREPPLine

5. Conclusiones y Trabajo Futuro

Hoy en día, debido a la creciente necesidad de obtener SI de alta calidad y con una productividad alta, el desarrollo basado en LPS se ha convertido en el enfoque de más éxito para asegurar la calidad, eficiencia económica y mantenibilidad de los SI [2]. Es por ello, y dada la complejidad y a la naturaleza extensiva de las LPS [9], que sea fundamental la incorporación de la ingeniería de requisitos de seguridad en las líneas de producto software, siendo mucho más importantes para la puesta en práctica del desarrollo basado en LPS, de lo que ya son para el desarrollo de un Sistema de Información (SI).

Debido a que los trabajos existentes que apuntan a especificar seguridad en líneas de producto, en los que se integre la perspectiva de la ingeniería de requisitos de seguridad, son escasos y no proporcionan soporte metodológico y sistemático para la gestión de la seguridad en LPS basada en la ingeniería de requisitos de seguridad y en los estándares de seguridad internacionales más importantes. En este artículo se presenta la aplicación en la práctica de un proceso sistemático que ayuda a desarrollar líneas de producto software seguras mediante la gestión integral de los requisitos de seguridad desde las primeras fases del ciclo de desarrollo y apoyándose en los estándares de seguridad internacionales más importantes (como ISO/IEC 15408 e ISO/IEC 21827; ISO/IEC 17799:2005, secciones: 0.3, 0.4, 0.6 y 12.1; ISO/IEC 27001, secciones: 4.2.1, 4.2.3, 4.3, 6.a y A.12.1.1), con el objeto de aportar una perspectiva que permita mejorar la calidad, tanto en las líneas de productos software como en los productos de dicha línea, los cuales serán conformes a dichos estándares.

Asimismo, a raíz de la realización de este caso de estudio presentado anteriormente podemos destacar las siguientes lecciones aprendidas más importantes:

- La aplicación de este caso de estudio nos ha permitido mejorar y refinar varias actividades de SREPPLine así como el modelo de variabilidad de seguridad y por tanto el repositorio de recursos de seguridad.
- El soporte de una herramienta es crucial para la aplicación práctica de este proceso en LPS de gran magnitud con mayor complejidad debido al número de artefactos manejados y las complejas relaciones de trazabilidad y gestión de la variabilidad de la línea que se tiene que realizar.
- Avanzar en el refinamiento del prototipo de herramienta CARE (Computer Aided Requierements Engineering) que estamos desarrollando, denominada SREPPLineTool, para avanzar en el nivel de automatización de SREPPLine.
- En lo relativo a los beneficios obtenidos por la Organización en la que se ha realizado el caso de estudio, se ha conseguido tener un proceso normalizado específico que sistematice la gestión de requisitos de seguridad en LPS y que facilite la conformidad de sus sistemas con la ISO/IEC 15408 e ISO/IEC 27001, así como la creación de un repositorio de recursos de seguridad cuyos artefactos serán reutilizables para el desarrollo de los sistemas que se deriven de la LPS o para el desarrollo futuro de otras LPS en la Organización.

Por último, hay una serie de aspectos planeados para el futuro, como es el refinamiento a partir de la realización de más casos de estudio del modelo teórico así como del prototipo de la herramienta (SREPPLineTool) que estamos desarrollando para automatizar la utilización de SREPPLine y nos permita incrementar el nivel de

automatización de la aplicación de SREPPLine y mejorar así la eficiencia del proceso de ingeniería de requisitos de seguridad de LPS.

Agradecimientos

Este artículo es parte del proyecto ESFINGE (TIN2006-15175-C05-05) del Ministerio de Educación y Ciencia, y de los proyectos MISTICO (PBC-06-0082) y DIMENSIONS (PBC-05-012-2) de la Consejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla- La Mancha y el FEDER.

Referencias Bibliográficas

1. Arciniegas, J.L., Dueñas, J.C., Ruiz, J.L., Cerón, R., Bermejo, J., and Oltra, M.A., *Architecture Reasoning for Supporting Product Line Evolution: An Example on Security*, in *Software Product Lines: Research Issues in Engineering and Management*, Käkölä, T. and Dueñas, J.C., Editors. 2006, Springer.
2. Birk, A., Heller, G., John, I., MaBen, T.v.d., Müller, K., and Schmid, K., *Product line engineering industrial nuts and bolts*. 2003, Fraunhofer IESE: Kaiserslautern.
3. Bosh, J., *Design & Use of Software Architectures*. 2000: Pearson Education Limited.
4. Clements, P. and Northrop, L., *Software Product Lines: Practices and Patterns*. SEI Series in Software Engineering. 2002: Addison-Wesley.
5. Faegri, T.E. and Hallsteinsen, S., *A Software Product Line Reference Architecture for Security*, in *Software Product Lines: Research Issues in Engineering and Management*, Käkölä, T. and Dueñas, J.C., Editors. 2006, Springer.
6. Firesmith, D.G., *Security Use Cases*. Journal of Object Technology, 2003: p. 53-64.
7. Giorgini, P., Massacci, F., Mylopoulos, J., and Zannone, N. *ST-Tool: A CASE Tool for Security Requirements Engineering*. in *IEEE International Conference on Requirements Engineering (RE'05)*. 2005.
8. Immonen, A., *A Method for Predicting Reliability and Availability at the Architecture Level*, in *Software Product Lines: Research Issues in Engineering and Management*, Käkölä, T. and Dueñas, J.C., Editors. 2006, Springer.
9. Käkölä, T. and Dueñas, J.C., *Software Product Lines: Research Issues in Engineering and Management*. 2006: Springer.
10. Kim, J., Kim, M., and Park, S., *Goal and scenario bases domain requirements analysis environment*. The Journal of Systems and Software, **79**(7) (2005). p. 926 - 938.
11. Kim., H.-K., *Automatic Translation Form Requirements Model into Use Cases Modeling on UML*. ICCSA 2005, LNCS, 2005: p. 769-777.
12. Kotonya, G. and Sommerville, I., *Requirements Engineering Process and Techniques*. Hardcover ed. 1998, UK: John Willey & Sons. 294.

13. López, F., Amutio, M.A., Candau, J., and Mañas, J.A., *Methodology for Information Systems Risk Analysis and Management*. 2005: Ministry of Public Administration.
14. McDermott, J. and Fox, C. *Using Abuse Case Models for Security Requirements Analysis*. in *Annual Computer Security Applications Conference*. 1999. Phoenix, Arizona.
15. Mead, N.R. and Stehney, T. *Security Quality Requirements Engineering (SQUARE) Methodology*. in *Software Engineering for Secure Systems (SESS05), ICSE 2005 International Workshop on Requirements for High Assurance Systems*. 2005. St. Louis.
16. Mellado, D., Fernandez-Medina, E., and Piattini, M., *SREPPLine: Towards a Security Requirements Engineering Process for Software Product Lines*. 9th International Conference on Enterprise Information Systems (ICEIS 2007). 5th International Workshop on Security In Information Systems (WOSIS-2007), 2007: p. 220-232.
17. Mellado, D., Fernández-Medina, E., and Piattini, M., *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), Springer LNCS 3982, 2006. **3**: p. 1044-1053.
18. Mellado, D., Fernández-Medina, E., and Piattini, M., *A Common Criteria Based Security Requirements Engineering Process for the Development of Secure Information Systems*. Computer Standards and Interfaces, 2007. **29**(2): p. 244 - 253.
19. Mouratidis, H. and Giorgini, P., *Integrating Security and Software Engineering: Advances and Future Visions*. 2007: Idea Group Publishing.
20. Pohl, K., Böckle, G., and Linden, F.v.d., *Software Product Line Engineering. Foundations, Principles and Techniques*. 2005, Berlin Heidelberg: Springer.
21. Popp, G., Jürjens, J., Wimmel, G., and Breu, R., *Security-Critical System Development with Extended Use Cases*. 2003: 10th Asia-Pacific Software Engineering Conference. p. 478-487.
22. Schmid, K., Krennrich, K., and Eisenbarth, M., *Requirements Management for Product Lines: A Prototype*. 2005, Fraunhofer IESE.
23. Sindre, G. and Opdahl, A.L., *Eliciting security requirements with misuse cases*. Requirements Engineering 10, 2005. **1**: p. 34-44.
24. Toval, A., Nicolás, J., Moros, B., and García, F., *Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach*. Requirements Engineering, **6**(4) (2002). p. 205-219.
25. Walton, J.P., *Developing a Enterprise Information Security Policy*. 2002, ACM Press: Proceedings of the 30th annual ACM SIGUCCS conference on User services.