PATROCINADORES

ugr Universidad de Granada

ATC

Sun microsystems

IBM

Bull

Ayuntamiento de Granada

cibersur

AEPIA
Asociación Española para la Inteligencia Artificial

AERFAI
Asociación Española de Reconocimiento de Formas y Análisis de Imágenes.

AIPO http://gdhis.unil.es:8080/aipo
Asociación Interacción Persona-Ordenador.

EUROGRAPHICS http://www.eg.org
Capítulo Español de la European Association for Computer Graphics.

EUSFLAT http://www.eusflat.org/index.htm
European Society for Fuzzy Logic and Technology.

SC of the IEEE CIS
Capítulo Español de la IEEE Computational Intelligence Society.

RADISC
Red Andaluza en Sistemas Complejos.

SEPLN http://www.sepln.org
Sociedad Española para el Procesamiento del Lenguaje Natural.

TIN - MEC
Programa Nacional de I+D en Tecnologías Informáticas. Ministerio de Educación y Ciencia.

CEA-IFAC http://www.cea-ifac.es
Comité Español de Automática de la International Federation of Automatic Control.

ISTANET http://www.istanet.net
Red Andaluza de Tecnología de Sistemas Inteligentes.

W3C http://www.w3c.es
Consorcio World Wide Web. Oficina Española.

RADI-AEB
Red Andaluza de Algoritmos Evolutivos y Bioinspirados.

ISBN: 84-9732-434-X

CEDI 2005 X Jornadas de Ingeniería del Software y Bases de Datos [JISBD'2005]

THOMSON

CEDI 2005

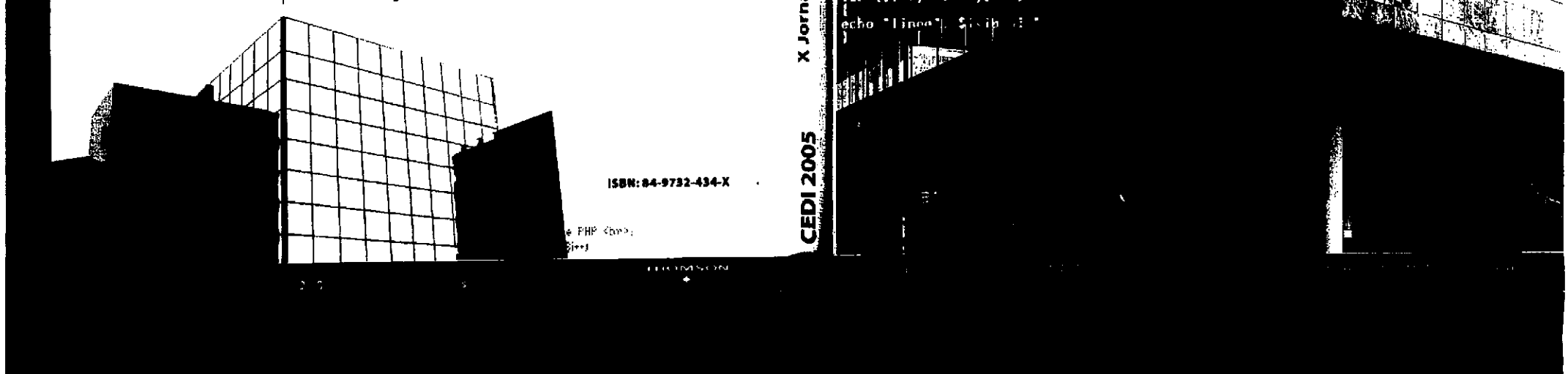
I CONGRESO ESPAÑOL DE INFORMÁTICA

GRANADA DEL 13 AL 16 DE SEPTIEMBRE

X Jornadas de Ingeniería del Software y Bases de Datos

[JISBD'2005]

EDITORES
Pedro Pablo Tovar Álvarez · Juan Hernández Núñez



ACTAS DE LAS

X Jornadas de Ingeniería del Software y Bases de Datos

[JISBD'2005]

EDITORES
Ambrosio Toval Álvarez
Juan Hernández Núñez

JORNADAS ORGANIZADAS POR
Sociedad de Ingeniería del Software y Tecnologías de Desarrollo de Software

THOMSON

Actas de las X Jornadas de Ingeniería
del Software y Bases de Datos [JISBD'2005]
© Los Autores

Editores de la serie de Actas del CEDI
Rafael Molina Soriano
Antonio Díaz García
Alberto Prieto Espinosa

Editores de las Actas de las presentes Jornadas
Ambrosio Toval Álvarez
Juan Hernández Núñez

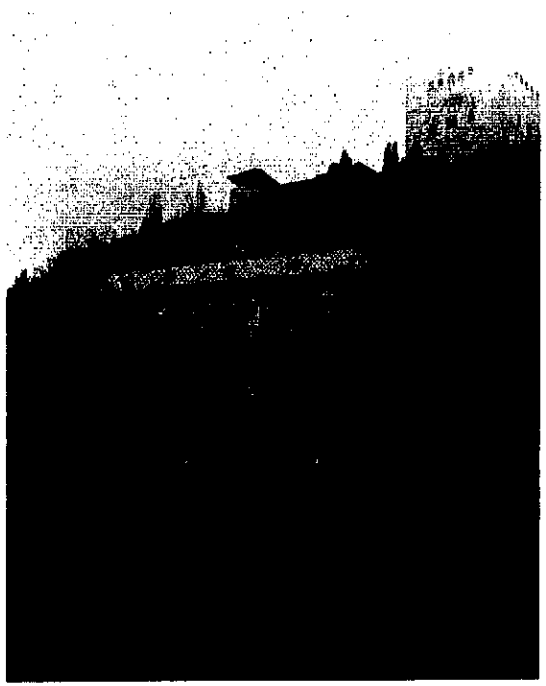
Diseño de Cubiertas
dixi
www.dhd-e.com

Impresión
THOMSON

COPYRIGHT© 2005 International
Thomson Editores Spain
Paraninfo, S.A.
Magallanes 25 · 28015 Madrid España
Tel: 91 446 33 50 · Fax: 91 445 62 18
clientes@paraninfo.es

Impreso en España
Printed in Spain

ISBN: 84-9732-434-X
Depósito legal: SE-4046-2005 European Union
Printed by Publidisa



Antiguo Hotel Reuma. Granada
Foto realizada para el CEDI2005 por DIXI

X JORNADAS DE INGENIERÍA DEL SOFTWARE Y BASES DE DATOS

COMITÉ EJECUTIVO

Presidente Comité de Programa	Ambrosio TOVAL (Universidad de Murcia)
Secretario Comisión Permanente	Mario PIATTINI (Universidad de Castilla-La Mancha)
Organización y Relaciones con CEDI'2005	Buenaventura CLARES (Universidad de Granada) Patricia PADEREWSKI (Universidad de Granada)
Coordinador de Tutoriales	José L. FERNÁNDEZ (Universidad de Murcia)
Coordinadora de Talleres	Patricia PADEREWSKI (Universidad de Granada)
Coordinador de Demostraciones	Francisco L. GUTIÉRREZ (Universidad de Granada)

PRESIDENCIA DEL COMITÉ PROGRAMA

Ambrosio TOVAL — Universidad de Murcia

COMITÉ DE PROGRAMA

J. ALDANA (U. Malaga)
B. ÁLVAREZ (U. Politécnica Cartagena)
J. ARAUJO (U. Nova de Lisboa)
M. J. ARAMBURU (U. Castellon)
O. BELO (U. do Minho)
P. BOTELLA (U. Politécnica Cataluña)
N. BRISABOA (U. de La Coruña)
C. CALERO (Castilla-La Mancha)
C. CANAL (U. de Málaga)
J.M. CAVERO (U. Rey Juan Carlos)
M. CELMA (U. P.Valencia)
R. CORCHUELO (U. de Sevilla)
Y. CRESPO (U. de Valladolid)
C. DELGADO (U. Carlos III)
O. DÍAZ (U. Politécnica País Vasco)
J. FALCÃO e CUNHA (U. do Porto)
X. FRANCH (U. Politécnica Cataluña)
P. de la FUENTE (U. de Valladolid)
L. FUENTES (U. de Málaga)
M. J. GASPAR da Silva (U. de Lisboa)
J. GÓMEZ (U. de Alicante)
J. HERNÁNDEZ (U. de Extremadura)
J. ITURRIOZ (U. Politécnica País Vasco)
N. JURISTO (U. Politénica de Madrid)
A. LOPES (U. de Lisboa)
H. MADEIRA (U. de Coimbra)
E. MARCOS (U. Rey Juan Carlos)
J.M. MARQUÉS (U. de Valladolid)
E. MENA (U. Zaragoza)
A. M. MORENO (U. Politénica de Madrid)
J. J. MORENO (U. Politécnica Madrid)
J. M. MURILLO (U. de Extremadura)
N.J. NUNES (U. Madeira)
O. PASTOR (U. Politécnica Valencia)
E. PIMENTEL (U. de Málaga)
A.POLO (U. Extremadura)
C. RAMOS (U. do Algarve)
I. RAMOS (U. Politécnica Valencia)
J. RIQUELME (U. Sevilla)
A. RITO (U. Técnica de Lisboa)
M.J. RODRÍGUEZ (U. Granada)
F. RUIZ (Castilla-La Mancha)
J. SAMOS (U. Granada)
F. SÁNCHEZ (U. de Extremadura)
J. SÁNCHEZ (U. Politécnica Valencia)
S. SOUSA BRITO (I. Politécnico Beja)
E. TENIENTE (U. P. Cataluña)
M. TORO (U. de Sevilla)
J. C. TRUJILLO (U. de Alicante)
B. VELA (U. Rey Juan Carlos)
A. VALLECILLO (U. de Málaga)

REVISORES ADICIONALES

Alberto ABELLÓ	Ma. Valeria de CASTRO
Alfredo GOÑI	Manuel RESINAS
Andre L. SANTOS	María del Mar RODÁN
Ángel HERRANZ	María Luisa RODRÍGUEZ
Antonio Cesar GÓMEZ	María-Isabel SÁNCHEZ-SEGURA
Antonio RUIZ	Marta RUIZ
Arantza ILLARRAMENDI	Miguel Ángel LAGUNA
Artur BORONAT	Nathalie MORENO
Bruno MARTINS	Nelson MEDINILLA
Carlos E. CUESTA	Noelia MAYA
Carme QUER	Norberto FERNÁNDEZ
Daniel GOMES	Nuria MEDINA
Enric MAYOL	Oscar DIESTE
Fran J. RUIZ-BERTOL	Pablo FERNÁNDEZ
Francisco COUTO	Paloma CÁCERES
Francisco Luis GUTIÉRREZ	Patricia PADEREWSKI
Ismael Navas DELGADO	Patricio LETELIER
Javier MUÑOZ	Pedro J. CLEMENTE
Jennifer PÉREZ	Pedro J. MUÑOZ
Joan Antoni PASTOR	Pedro SÁNCHEZ
Joao Pedro NETO	Pedro VALDERAS
José Luis GARRIDO	Rafael BERLANGA
José Miguel BLANCO	Raul ROMERO
José Miguel CAÑETE	Sira VEGAS
José Ramon RIOS	Toni URPÍ
Juan Ángel PASTOR	Vicente LUQUE
Juan Manuel VARA	Vicente PELECHANO
Luis SÁNCHEZ	Xavier FERRÉ

COORDINADOR DE TALLERES

Patricia PADEREWSKI	Universidad de Granada

MIEMBROS COMITÉ DE TALLERES

Mª Visitación HURTADO	Universidad de Granada
Juan Manuel MURILLO	Universidad de Extremadura
José SÁEZ	Universidad de Murcia
Mª Dolores LOZANO	Universidad de Castilla La Mancha
José Hilario CANÓS	Universidad de Valencia
Cecilia DELGADO	Universidad de Granada
Amador DURÁN	Universidad de Sevilla

COORDINADOR DE TUTORIALES

José L. FERNÁNDEZ	Universidad de Murcia

COORDINADOR DE DEMOSTRACIONES

Francisco L. GUTIÉRREZ	Universidad de Granada

SISTEMA AUTOMÁTICO DE REVISIÓN

Quercus Software Engineering Group

Pablo AMAYA	Universidad de Extremadura
Jose M. CONEJERO	Universidad de Extremadura

CONTENIDOS

Artículos.....**1**

Um Quadro de Referência para a Comparação de Metodologias Ágeis.....3
Joao Carlos Ribeiro, Joao Araujo

Búsqueda Tabú para la generación de casos de prueba de cobertura de bucles.11
María Eugenia Díaz Fernández, Raquel Blanco, Javier Tuya

Providing platforms for developing pervasive systems with MDA. An OSGi metamodel.....19
Javier Muñoz, Vicente Pelechano, Estefanía Serral

PRISMANET middleware: Soporte a la Evolución Dinámica de Arquitecturas Software Orientadas a Aspectos.....27
Cristóbal Costa Soria, Jennifer Pérez, Nour Ali, José Á. Carsí, Isidro Ramos

Sistematizando la Especificación de Requisitos Safety: un Caso de Estudio sobre Aplicaciones Teleoperadas.....35
Elena Navarro, Pedro Sánchez, Patricio Letelier, Juan A. Pastor, Isidro Ramos

Una Arquitectura para la Integración de Portales Web basada en Servicios Web Semánticos.....43
César J. Acuña, Juan M. Gómez, Esperanza Marcos, Christoph Bussler

Producción Científica en Ingeniería de Requisitos en España: Un Análisis en el Contexto Europeo.....51
Oscar Dieste, Natalia Juristo, Ana M. Moreno, Alan M. Davis, Ann Hickey

Soporte de Métricas con Independencia del Lenguaje para la Inferencia de Refactorizaciones.....59
Raúl Marticorena Sánchez, Yania Crespo González-Carvajal, Carlos López Nozal

Supporting the Automatic Generation of Advanced Modelling Environments with Graph Transformation Techniques.....67
Esther Guerra, Paloma Díaz, Juan de Lara

Un servicio web de políticas de acceso basadas en roles para hipermedia.....75
Daniel Sanz García, Ignacio Aedo, Paloma Díaz

Síntesis de patrones de interacción a partir de diagramas de secuencia en UML.....83
Miguel Ángel Pérez, Amparo Navasa Martínez, Juan Manuel Murillo, Carlos Canal Velasco

Modelos estructurales de aspectos para arquitectura de software.....91
Carlos E. Cuesta, M. Pilar Romay, Pablo de la Fuente, Manuel Barrio Solórzano

Finding where to apply object-relational database schema refactorings: an ontology-guided approach.....99
Coral Calero Muñoz, Aline Baroni, Fernando Brito e Abreu

Do composite states improve the understanding of UML statechart diagrams?.....107
José Antonio Cruz Lemus, Marcela Genero, Esperanza Manso, Mario Piattini

Transformaciones MDA sobre especificaciones computacionales de UML 2.0 a Maude.....115
José Raúl Romero Salguero, Nathalie Moreno, Antonio Vallecillo

Improving automatic SQL translation for ROLAP tools.....123
Oscar Romero Moral, Alberto Abelló Gamazo

A Hybrid Method for Discovering Distance-Enhanced Inter-Transactional Rules.....131
Pedro Gabriel Ferreira, Ronnie Alves, Paulo Azevedo, Orlando Belo

The Effect of Coupling on Understanding and Modifying OCL Expressions: An Experimental Analysis.....139
Luis Reynoso, Marcela Genero, Mario Piattini, Esperanza Manso

Generación Automática de Aplicaciones Mixtas Sw/Hw mediante la Integración de Componentes COTS.....147
Cristina Vicente Chicote, Ana Toledo Moreo, Carlos Fernández Andrés, Pedro Sánchez

Método de unión de modelos independientes de plataforma en MDA.....155
Álvaro Prieto Ramos, Adolfo Lozano-Tello, Encarna Sosa Sánchez

A product-line approach to database reporting.....163
Felipe I. Anfurrutia, Oscar Diaz, Salvador Trujillo

Un Enfoque Orientado a Procesos para la Especificación de Planes de Emergencia.....171
Manuel Llavador, Patricio Letelier, Marcos R. S. Borges, José H. Canós, Mª Carmen Penadés, Carlos Solís

De la Arquitectura Software al Urbanismo Software: Hacia Nuevas Formas de concebir los Sistemas de Software Intensivo.....179
Juan José Moreno-Navarro

Adaptación de las normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la evaluación de la madurez de procesos software en países en desarrollo.....187
Francisco J. Pino, Felix Garcia, Francisco Ruiz, Mario Piattini

Un entorno integrado para la reingeniería.....195
Ignacio García Rodríguez de Guzmán, Macario Polo Usaola, Mario Piattini

PWSSEC: Proceso de Desarrollo para Seguridad de Servicios.....203
Carlos Gutiérrez García, Eduardo Fernández-Medina, Mario Piattini

Medidas de Usabilidad de Componentes Software.....211
Manuel F. Bertoa, Antonio Vallecillo

ORCDB: Arquitectura para la extensión de la semántica de SQL en bases de datos restrictivas orientadas a objetos con restricciones polinómicas de igualdad.....221
M. Teresa Gómez-López, Rafael M. Gasca, Carmelo Del Valle, Victor Cejudo

Determinación de los requerimientos de calidad del producto software basados en normas internacionales.....231
Abraham Eliseo Dávila Ramón, Karin Ana Melendez Llave, Luis Alberto Flores García

Artículos Cortos.....239

Una aproximación metodológica para soportar la evolución de requisitos a partir de un modelo arquitectónico OA.....241
Amparo Navasa Martínez, Miguel Ángel Pérez, Juan Manuel Murillo

Mejorando la accesibilidad de las aplicaciones GIS basadas en Web.....247
Miguel R. Luaces, Nieves R. Brisaboa, Jose R. Parama, David Trillo, Jose R. R. Viqueira

Del método formal a la aplicación industrial en Gestión de Modelos: Maude aplicado a Eclipse Modeling Framework.....253
Artur Boronat, José Iborra, José Á. Carsí, Isidro Ramos, Abel Gómez

Análisis de los Métodos de Selección de Componentes COTS desde una Perspectiva Ágil.....259
Fredy Javier Navarrete Ramirez, Pere Botella, Xavier Franch

Un Profile para el Modelado de Patrones de Software.....265
José Luis Isla Montes, Francisco Luis Gutiérrez Vela, Patricia Paderewski Rodríguez

Recuperación del conocimiento basada en contexto: Una aplicación en la Arqueología (ArqueOnto).....271
Juan María Fernández González, Antonio Polo Márquez, Luis Jesús Arévalo Rosado, Enrique Cerrillo Cuenca

Desarrollando aplicaciones hipermedia para la Web Semántica.....277
Laura Montells Higuero, Susana Montero, Paloma Díaz, Ignacio Aedo

Arquitectura para la Clasificación y Composición de Servicios Web.....283
Ismael Navas Delgado, Maria del Mar Rojano-Muñoz, Jose F. Aldana-Montes

Diagramas de casos de uso para el análisis de requisitos en almacenes de datos.....289
Jose Norberto Mazón López, Juan Trujillo, Manuel Serrano, Mario Piattini

Especificación de jerarquías de dimensión en un almacén de datos usando WordNet.....295
Jose Norberto Mazón López, Juan Trujillo, Manuel Serrano, Mario Piattini

PWSSEC: Proceso de Desarrollo para Seguridad de Servicios Web

Carlos A. Gutiérrez[1], Eduardo Fernández-Medina[2], Mario Piattini[2]

(1) STL, Madrid, c_gutierrez@acm.org
(2) Grupo de Investigación Alarcos. Universidad de Castilla-La Mancha.
Paseo de la Universidad 4, 13071, Ciudad Real. Tel: 34 926 29 53 00
{Eduardo.FdezMedina, Mario.Piattini}@uclm.es

Resumen

Los servicios Web (WS) se han consolidado en los últimos años como el middleware estándar sobre el que integrar sistemas heterogéneos y llevar a cabo complejos procesos de negocio interorganizacionales. En los últimos años se han desarrollado un gran número de estándares basados en WS algunos de ellos ofreciendo soluciones estándar en lo relativo a su seguridad. Es tal la magnitud del número de estándares de seguridad producidos que la carencia de una aproximación global dificulta su aplicación práctica. En este artículo se presenta el proceso de desarrollo PWSSec (Proceso de Desarrollo de Servicios Web Seguros) cuyo propósito es dotar de seguridad a un sistema software basado en WS teniendo en cuenta este aspecto en todas y cada unas de las etapas del desarrollo del sistema software. Este proceso es aplicable tanto en proyectos de desarrollo software basado en WS que se inician desde cero como en sistemas en los que la arquitectura funcional ya está diseñada. El objetivo final de este proceso es obtener una especificación de los requisitos y la arquitectura de seguridad específica de WS que se integre con el resto de la arquitectura funcional y no funcional, y que esté basada en estándares para WS.

1. Introducción

El paradigma de los WS ha alcanzado tal relevancia en el mundo académico y en la industria que la visión de Internet está evolucionando pasando de ser considerada como un mero repositorio de datos estáticos para convertirse en la infraestructura base y fundamental sobre la que llevar a cabo complejos procesos de negocio y alianzas [23].

Una prueba de esta relevancia se confirma con el informe publicado en Junio del 2004 por el IDC [12] en el que se estima que el mercado de soluciones basadas en WS crecerá de manera estable hasta alcanzar los 11 billones de dólares en el 2008.

La seguridad en los WS es un aspecto clave para su completa aceptación como paradigma de integración interorganizacional por defecto. De hecho, en los últimos años los consorcios más relevantes vinculados con la producción de estándares en Internet y en WS, como son el IETF, el W3C o OASIS, han dedicado un gran esfuerzo a la seguridad en dicho paradigma.

Pese a este espectacular crecimiento, todavía no existe un proceso de desarrollo como PWSSec que facilite la integración sistemática de la seguridad en todas las etapas del desarrollo de sistemas software basados en WS. Por tanto la principal contribución realizada en este artículo es la descripción del proceso PWSSec.

En la siguiente sección se presenta una visión general del proceso presentando los principios y fundamentos aplicados en PWSSec; en la sección 3 se desarrolla la etapa WSSecReq (Requisitos de Seguridad para Servicios Web) que se ocupa de la especificación de los requisitos de seguridad específicos para WS; en la sección 4 se describe la etapa WSSecArch (Arquitectura de Seguridad para Servicios Web) que se ocupa de especificar la arquitectura de servicios de seguridad que cubre los requisitos de seguridad procedentes de la etapa WSSecReq; en la sección 5 se presenta la etapa WSSecTech (Tecnologías de Seguridad para

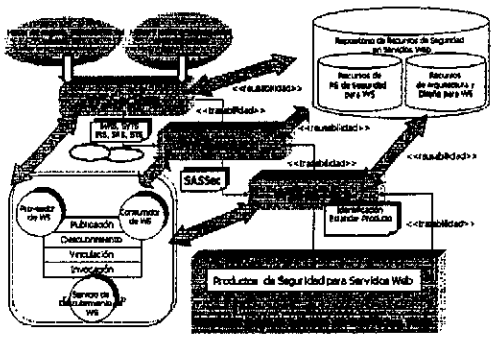


Figura 1. Esquema general del proceso PWSSec.

Servicios Web) en la que se definen las tecnologías de seguridad, en forma de estándares, que realizan los servicios de seguridad identificados en la etapa WSSecArch; en la sección 6 se presentan brevemente dos casos de estudio de proyectos reales en los que se está aplicando este proceso y, por último, en la sección 7 se ofrecen un conjunto de conclusiones.

2. Visión general de PWSSec

El proceso PWSSec permite definir los requisitos de seguridad para sistemas basados en WS y describe una arquitectura de seguridad de referencia que facilita el diseño e implementación de arquitecturas concretas de seguridad basadas en WS que implementen cierto conjunto de estándares.

En general, las principales características de este proceso son: i) Proceso iterativo e incremental de forma que facilita el desarrollo y la gestión de los riesgos [4] y la integración gradual de la seguridad en los sistemas basados en WS [5]; ii) trazabilidad y reusabilidad del proceso de desarrollo e interoperabilidad y reusabilidad de los productos; iii) proceso centrado en los elementos y los procedimientos básicos definidos para una arquitectura basada en WS [21]: los actores básicos son los agentes proveedores de servicios, los agentes consumidores de los servicios y los agentes de descubrimiento mientras que los procedimientos básicos son publicación, descubrimiento, vinculación e invocación.

La Figura 1 muestra las etapas en la cuales se encuentra estructurado el proceso PWSSec.

3. WSSecReq (Requisitos de Seguridad para Servicios Web)

El principal objetivo de esta etapa es producir de manera sistemática una especificación de los requisitos de seguridad de un sistema basado en WS. Esta etapa consta de las siguientes actividades: elicitación, especificación, análisis y verificación y validación de los requisitos de seguridad software.

La reusabilidad y trazabilidad de los requisitos de seguridad y del trabajo de elicitación y análisis, se consigue utilizando el siguiente conjunto de artefactos de una forma razonada y relacionada: i) Árboles de amenazas abstractos y concretos [15, 17] asociados a cada patrón de negocio y de aplicación para WS definidos en [6]; ii) escenarios de ataque abstractos y concretos, definidos como casos de mal uso según [1, 18], representados en base al perfil QoS UML [16] y agrupados en perfiles de ataque [15]; iii) casos de uso de seguridad según D. G. Firesmith, SG. Sindre y A.L. Opdhal [7]; iv) plantillas reutilizables de requisitos de seguridad específicos de WS [9]; v) y una especificación formal de los requisitos de seguridad basada en SIREN [19].

La trazabilidad de los requisitos consiste en identificar, a partir del servicio o funcionalidad bajo análisis (ej: caso de uso), el patrón de negocio WS y su correspondiente patrón de aplicación WS [6]. Para cada uno de estos tipos de patrones se ha definido un árbol abstracto de amenazas que se deberá instanciar para obtener un árbol de amenazas específico para la

funcionalidad bajo análisis. Ambas instancias del árbol, de negocio y de aplicación, se combinan para obtener un único árbol de amenazas del sistema. Además, para cada patrón de aplicación de WS se ha definido un perfil de ataque que agrupa un conjunto de casos de mal uso abstractos que deberán ser instanciados para obtener un conjunto de ataques específicos que lleven a cabo las amenazas definidas por el patrón de aplicación. Cada caso de mal uso abstracto tiene asociado uno o más casos de uso de seguridad abstractos que, al ser instanciados, describen el comportamiento que debe tener el sistema para contrarrestar el ataque. A su vez, cada caso de uso de seguridad abstracto tiene asociado un conjunto de una o más plantillas de requisitos de seguridad específicas para WS que deberán ser instanciadas para obtener el conjunto de requisitos de seguridad, específicos de WS, de la funcionalidad analizada.

Todo este conjunto de artefactos empleados como soporte para desarrollar las actividades de esta etapa son incluidos en el repositorio Recursos de RE de Seguridad para WS (ver Figura 1) que se actualiza constantemente.

3.1. Entrada y Salida

La entrada de la etapa WSSecReq está compuesta de: i) Una especificación del alcance funcional que se desea abarcar en la iteración junto con los riesgos de seguridad definidos en iteraciones anteriores (si las hubo) que tengan la prioridad más alta y que no hayan sido todavía resueltos; ii) las metas de negocio y seguridad definidas para el sistema; iii) la parte de la política de seguridad organizacional que estimamos pueda influenciar en el diseño del sistema.

Adoptado de SIREN [19], la salida de esta etapa será: i) Especificación de los Requisitos de Seguridad del Sistema (SyRS); ii) Especificación de los Requisitos de Seguridad Software (SRS); ii) Especificación de las Pruebas de Sistema (SyTS); iii) Especificación de las Pruebas Software (STS); iv) Especificación de los Requisitos de Seguridad de la Interfaz (IRS), extraídos de entre los SRS y para evitar que los documentos sean extremadamente grandes.

3.2. Actores

Los actores que toman parte de esta etapa son el conjunto de personas Involucradas, el Equipo de RE (ingeniería de requisitos) y el Equipo de Seguridad. Las personas Involucradas actuarán como fuentes de las metas, políticas y restricciones de negocio de más alto nivel. El Equipo RE llevará a cabo las actividades de elicitación, análisis, especificación y verificación y validación de los requisitos de seguridad conjuntamente con los miembros del Equipo de Seguridad.

3.3. Actividades

Elicitación

Hemos definido la actividad de elicitación en base a los siguientes pasos [8]:

1. Identificar el conjunto de WS funcionales a proteger.
2. Identificar los tipos de atacantes.
3. Identificar las posibles amenazas. En la Figura 2, se muestra cómo se ha utilizado el perfil UML QoS para modelar el nivel de calidad (mostrado con el esterotipo <<Asset>>) asociado con cierto WS de negocio denominado *InfoPacienteSistemaSanidadWS*.

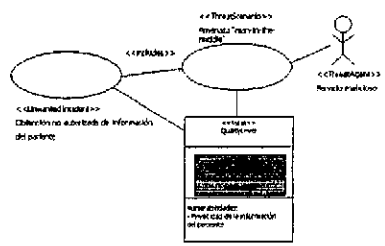


Figura 2. Modelado de los escenarios de amenaza y de los atacantes.

Además, se muestra un ejemplo en el que se emplea el perfil mencionado para modelar los incidentes inesperados (es decir, las amenazas). El activo 'QualityLevel' definido se relaciona con una amenaza potencial identificada que, a su vez, se encuentra relacionada con su atacante. Además, el activo 'QualityLevel'

presenta una relación con la amenaza originada por ataques de tipo 'man-in-the-middle'.
4. Evaluar el impacto de las amenazas.
5. Estimar y priorizar los riesgos de la seguridad.
6. Seleccionar los subfactores de seguridad [9] (p.e: autenticación, autorización, etc.) que determinen los tipos de requisitos de seguridad necesarios para reducir ese riesgo a un nivel aceptable.
7. Especificación del requisito de seguridad que abarca los siguiente pasos:

- Seleccionar del repositorio las plantillas para cada riesgo de seguridad y subfactor asociado. En nuestro caso, un ejemplo de plantilla para la privacidad de la información podría ser: "El [WS consumidor | WS proveedor | WS descubrimiento] garantizará la no revelación de [tipo | identificador] de información sin el consentimiento expreso de su propietario al [WS consumidor | WS proveedor | WS descubrimiento] durante la ejecución de [conjunto de interacciones | casos de uso | subsistema] según el criterio y medidas dado en la tabla [tabla]".
- Determinar el criterio de seguridad, de forma que se introduzcan sus parámetros en la plantilla. Ej.: "Peticiones que no disponen del nivel de autorización requerido que son rechazadas al solicitar el valor de ciertos atributos críticos".
- Determinar las métricas de seguridad adecuadas que midan la existencia de los criterios de seguridad escogidos e introducir la métrica de calidad en la plantilla. En nuestro caso la métrica se podría medir en tanto por ciento.
- En función al riesgo de seguridad identificado para cierto servicio, determinar el nivel mínimo aceptable de la métrica para el criterio escogido que limite a un nivel aceptable el riesgo asociado e introducir el nivel requerido en la plantilla. Por ejemplo, el 99.99%.
- Especificar el requisito de seguridad instanciando la plantilla a partir de los parámetros seleccionados en los tres últimos pasos. Un ejemplo de instancia de requisito de privacidad para un servicio de atributos que participa en una solución de federación podría ser:

"El WS proveedor de atributos del sistema serviciodesalud.ejemplo.com garantizará (tal y como se explica en la siguiente tabla) la no divulgación de la información personal considerada crítica de los pacientes frente a ataques sofisticados si éstos no han dado un consentimiento expreso que permita revelarlos al WS cliente GeneradorEstadísticas del sistema de la empresa XYZ durante la ejecución de los Casos de Uso Generación de Estadísticas y Previsiones".

	Número mínimo de atributos críticos cuya privacidad está garantizada
WS GeneradorEstadísticas genera un informe del Perfil del Paciente	99.99%
...	...
WS GeneradorEstadísticas genera un informe de predicción ocupacional.	99.99%

Tabla 1. Ejemplo de criterios para el requisito de privacidad de ejemplo.

Análisis
La actividad de análisis consiste básicamente en i) identificar los posibles conflictos que pudieran surgir entre los requisitos de seguridad derivados de escenarios de composición e integración; ii) clasificar los requisitos de seguridad en requisitos de sistema, de software o de interfaces [19].
Especificación
Consiste básicamente en documentar los requisitos de seguridad y se fundamenta en el uso de un conjunto de plantillas de documentos de requisitos y en una estructura de documentos reutilizables definida en SIREN [19].
Verificación y Validación
La última etapa definida en WSSecReq consiste en: i) Verificación interna que identifique los posibles conflictos entre los requisitos de seguridad y el resto de requisitos y que detecte especificaciones de requisitos incompletas, ambiguas o mal redactadas; ii) validación externa que se deberá realizar de manera conjunta con los participantes en esta etapa.

4. WSSecArch (Arquitectura de Seguridad para Servicios Web)

La etapa WSSecArch abarca el diseño de la arquitectura [2] de seguridad distribuyendo los requisitos de seguridad en mecanismos de seguridad (servicios) que permiten mitigar los riesgos identificados en la etapa anterior.

Los dos principios característicos de esta etapa son, al igual que lo eran en WSSecReq, la reusabilidad y la trazabilidad de los productos y del proceso. Las soluciones de seguridad arquitectónicas se toman a partir de patrones de arquitectura de seguridad que se mantienen en un repositorio (Recursos de Arquitectura y Diseño de Seguridad para WS) que incluye, el marco de razonamiento que los relaciona con cada uno de los subfactores de la seguridad [13]. El estudio del conjunto de especificaciones e iniciativas de investigación en el campo de la seguridad para WS [10] nos permite abstraer las características comunes a todas ellas creando este repositorio de patrones de arquitectura de los servicios de seguridad.

4.1. Entrada y Salida

Las entradas contempladas en esta etapa son: i) Metas del negocio de la iteración actual; ii) metas y políticas de seguridad organizacionales tenidas en cuenta durante la iteración actual; iii) el conjunto de escenarios de ataque y seguridad desarrollados en WSSecReq; iv) el conjunto de requisitos de seguridad definidos en las especificaciones SyRS, SRS, SyTS, STS, IRS desarrolladas en la etapa WSSecReq.

La salida de esta etapa debe ser una especificación de la arquitectura de seguridad (SASSec en la Figura 1) complementada con el conjunto de requisitos de seguridad resueltos por la arquitectura, conjuntamente con el conjunto de patrones de arquitectura de seguridad que los implementan y el catálogo de políticas de seguridad asociados con los servicios WS de negocio y de seguridad.

4.2. Actores

Los principales actores que intervienen en esta etapa son el equipo de RE, el Equipo de Diseño de la Arquitectura y el Equipo de Seguridad. En la etapa de Identificación de los Patrones de Arquitectura de Seguridad y de Integración de los Patrones Arquitectónicos de Seguridad intervienen los dos últimos mientras que en la etapa de Validación de la Arquitectura de Seguridad deben intervenir los tres.

4.3. Actividades

Identificación de los Patrones de Arquitectura de Seguridad
Para cada requisito de seguridad de cada servicio de negocio perteneciente a la iteración actual, se debe identificar el patrón de arquitectura de seguridad de WS que lo resuelve. Este patrón de arquitectura define un conjunto de tipos de servicios abstractos (puesto que no definen cómo deben ser implementados en términos de algoritmos o tipos de datos concretos) y de interacciones que especifican formalmente las propiedades de seguridad ofrecidas por el patrón de seguridad. Estos nuevos servicios ofrecerán nuevas funcionalidades de seguridad que deberán ser reevaluadas, analizadas y refinadas desde el proceso WSSecReq.
Definición de la Política de Seguridad asociada al Patrón.
Las políticas de seguridad permiten definir preferencias, requisitos y capacidades [20] a los servicios de seguridad abstractos y a los servicios de negocio. Cada servicio de seguridad abstracto, derivado de uno o más requisitos de seguridad a través de la aplicación de cierto patrón arquitectónico, deberá indicar en su política de seguridad, además de los posibles parámetros para su instanciación, el conjunto de tipos de requisitos de seguridad que cubre (p.ej. autenticación, disponibilidad, etc.).
Integración de los Patrones Arquitectónicos de Seguridad
Con el propósito de obtener un método sistemático que defina la arquitectura de seguridad, hemos elaborado una arquitectura de referencia de seguridad que refleja de manera directa la trazabilidad de los requisitos de seguridad con los correspondientes componentes software que los implementan. Aunque por motivos de espacio aquí se presenta de forma breve, esta arquitectura cubre tanto el aspecto operativo de la seguridad como el de su administración. Algunos de los elementos básicos manejados en la arquitectura de seguridad de referencia para WS son los siguientes:

- Zona de Seguridad: un área perteneciente a una federación de organizaciones, organización, departamento o área en la que se encuentran disponibles un conjunto de WS gobernados por una misma entidad. Una Zona contendrá uno o más Núcleos de Seguridad WSSecKern (ver siguiente punto) que

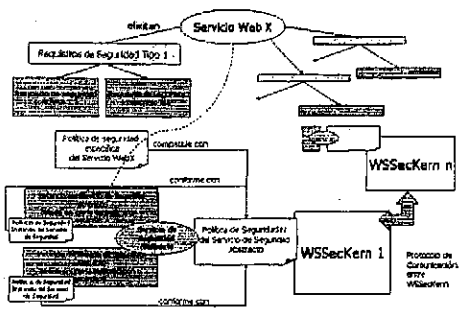


Figura 3. Arquitectura de seguridad de referencia de WS.

actuarán como pasarelas de seguridad para los WS disponibles en ella.

- Kernel de Seguridad de WS (WSSecKern), es el corazón de la arquitectura de referencia y se responsabiliza de gestionar un conjunto de Servicios de Seguridad, derivados de la aplicación de cierto conjunto de patrones de arquitectura de seguridad, y destinados a servir a un conjunto potencial de servicios de negocio. Cada WSSecKern soportará uno o más Servicios de Seguridad Abstractos implementados mediante uno o más mecanismos de seguridad concretos en forma de estándares o especificaciones.
- Servicio de Seguridad Abstracto, que cubre cierto conjunto de tipos de requisitos de seguridad y que puede poseer diversas instancias en función al número de implementaciones, basadas en estándares de seguridad para WS, que se definan en WSSecTech.
- Políticas de Seguridad de un Servicio de Seguridad Abstracto: incluye los posibles parámetros o atributos con los que se deben definir las políticas de seguridad de las potenciales instancias del Servicio de Seguridad Abstracto así como una descripción del conjunto de tipos de requisitos de seguridad que maneja el Servicio de Seguridad Abstracto.
- Política de Seguridad de una Instancia de Servicio de Seguridad Abstracto que define las capacidades soportadas por el estándar empleado. Esta política no se definirá hasta que se seleccionen los estándares de seguridad en la etapa WSSecTech.
- Política de Seguridad de Servicio de Negocio: definida por cada WS de negocio y registrada en el WSSecKern cuando se da de alta en éste. El servicio de negocio define qué conjunto de requisitos de seguridad desea y qué mecanismo, y cómo desea utilizarlo (ej.: "quiero autenticación simple del mensaje basada en certificados X.509v3").
- Protocolo de Intercomunicación entre los WSSecKern: permite la coordinación e interacción de las diferentes Instancias de los Servicios de Seguridad registradas en cada uno de ellos.

Validación de la Arquitectura de Seguridad

Consiste básicamente en validar que los escenarios de amenaza y de seguridad para la iteración en curso son resueltos[2].

Especificación de la Arquitectura de Seguridad

Esta actividad plasma en un documento de Especificación de Arquitectura de la Seguridad (SAS en la figura 1) mediante el uso de vistas [3, 14] que muestren cómo se despliegan los escenarios de seguridad, mediante las interacciones de los componentes de la arquitectura (WSSecKern y sus Servicios de Seguridad Abstractos, Agentes WS Consumidores y Agentes WS Proveedores), como contramedida de los escenarios de ataque contemplados en la iteración actual.

5. WSSecTech (Tecnologías de Seguridad de Servicios Web)

Uno los aspectos clave que justifican la gran popularidad de los WS es su inherente alto grado de interoperabilidad. Esta interoperabilidad está garantizada por el uso exclusivo de estándares aprobados por los consorcios más importantes relacionados con las tecnologías de Internet y de WS (IETF, W3C, OASIS, WS-I, Liberty Alliance Project, etc.). El objetivo de esta etapa es identificar los estándares de seguridad necesarios para realizar la arquitectura definida en el SASSec que cumplan la política de negocio y de seguridad así como las posibles restricciones del sistema.

5.1. Entradas y Salidas

Esta etapa nos permitirá, a partir de los subsistemas e interacciones diseñadas en la arquitectura SASSec, y teniendo en cuenta las metas y los requisitos de negocio y de seguridad, obtener: i) Un catálogo de especificaciones y estándares de seguridad de WS; ii) una relación de tecnologías seleccionadas para implementar los distintos servicios (e.g.: implementar WSSecKern como un ESB, como un microkernel JMX, como manejadores Axis de Apache o como módulos de cierto sistema operativo); iii) un razonamiento que justifique la selección de cada especificación y estándar para cada WSSecKern.

5.2. Actores

Los principales actores deberán ser el Equipo de Diseño, que deberá estar especializado en los estándares y especificaciones de seguridad en WS, el Equipo de Diseño de Arquitectura como entidad de soporte y el Equipo de Seguridad supervisando cada actividad y resolviendo dudas como por ejemplo en el caso de un servicio de autenticación indicando si existe o no una infraestructura de autenticación (ej.: PKI, Kerberos, etc.) en la que se debe basar la selección de la oportuna especificación y/o estándar.

5.3. Actividades

Identificación de Especificación/Estándar

Esta actividad consiste en la selección e integración de un conjunto de estándares de WS que cumplan con el grado de interoperabilidad requerido. De nuevo, y con el propósito de promover la reusabilidad, esta etapa se apoya en un repositorio de información en el que están clasificados cada una de las iniciativas y cada una de las especificaciones y estándares desarrollados en el ámbito de la seguridad en WS.

Esta base de conocimiento cataloga estas especificaciones y estándares a partir de la asignación de una calificación cualitativa sobre ciertos criterios de clasificación como por ejemplo: sistema ligero/pesado, grado de aceptación, grado de implementación (e.g.: número de productos que lo han implementado), estado de evolución (e.g.: versiones liberadas), coste económico, curva de aprendizaje, coste de mantenimiento, adaptabilidad, reusabilidad, extensibilidad, interoperabilidad etc. Por ejemplo, una especificación o estándar incluida en el perfil WS-I Basic Security Profile [22] puntuará más alto en el apartado de interoperabilidad que otra que ofrezca una solución alternativa pero que no esté incluida.

Definir la Política de Seguridad de las Instancia del Servicio de Seguridad Abstracto

En esta etapa se deberá definir las políticas de seguridad de cada una de las Instancias de los Servicios de Seguridad en base al estándar que implementen.

6. Casos de estudio

En la actualidad, el proceso aquí presentado se está aplicando en dos casos de estudio. El primero de ellos nos permite estandarizar el flujo de trabajo de un sistema de transferencia bancaria [11] mientras que el segundo consiste encontrar una solución genérica, basada en un desarrollo a medida, y que sólo acepte clientes tipo Web, para autenticación y Single Sign-On. La aplicación de PWSSec en estos casos reales no está permitiendo refinarlo completando los artefactos reutilizables definidos en cada una de las etapas así como identificando aspectos donde aplicar mejoras.

7. Conclusiones

En este artículo hemos presentado PWSSec, un proceso para el desarrollo de sistemas seguros

basados en WS, que estamos refinando mediante su utilización en dos casos de estudio. Este proceso cubre las etapas de requisitos y diseño permitiendo obtener los requisitos, la arquitectura y los estándares de seguridad de un sistema basado en WS.

Agradecimientos

Este trabajo se ha realizado en el marco del proyecto CALIPO (TIC2003-07804-C05-03) y la red RETISTIC (TIC2002-12487-E), de la Dirección General de Investigación del Ministerio de Ciencia y Tecnología.

Referencias

[1] I. Alexander, "Misuse Cases: Use Cases with Hostile Intent", *IEEE Computer Software*, vol. 20, pp. 58-66, 2003.

[2] L. Bass and R. Kazman, "Architecture-Based Development", Carnegie Mellon. Software Engineering Institute. CMU/SEI-99-TR-007, April 1999.

[3] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*, 2nd, ed: Addison-Wesley, 2003.

[4] B. W. Boehm, "A Spiral Model of Software Development and Enhancement", *IEEE Computer*, pp. 61-72, 1988.

[5] R. Breu, K. Burger, M. Hafner, J. Jürjens, G. Popp, V. Lotz, and G. Wimmel, "Key Issues of a Formally Based Process Model for Security Engineering", Proc. ICSSEA03, 2003.

[6] M. Endrei, J. Ang, A. Arsanjani, S. Chua, P. Comte, P. Krogdahl, M. Luo, and T. Newling, *Patterns: Service-Oriented Architecture and Web Services*, 1st ed, 2004.

[7] D. G. Firesmith, "Security Use Cases", *Journal of Object Technology*, vol. 2, pp. 53-64, 2003.

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

[9] D. G. Firesmith, "Specifying Reusable Security Requirements", *Journal of Object Technology*, vol. 3, pp. 61-75, 2004.

[10] C. Gutiérrez, E. Fernández-Medina, and M. Piattini, "Web Services Security: is the problem solved?" *Information Systems Security*, vol. 13, pp. 22-31, 2004.

[11] C. Gutiérrez, E. Fernández-Medina, and M. Piattini, "Desarrollo de sistemas de servicios web seguros", Proc. JSWEB'05, Granada, Spain, 2005.

[12] IDC, "Cautious Web Services Software Adoption Continues; IDC Expects Spending to Reach $11 Billion by 2008", 2004.

[13] M. Klein and R. Kazman, "Attribute-Based Architectural Styles", Software Engineering Institute CMU/SEI-99-TR-022, October 1999.

[14] P. Krutchen, "The 4+1 View Model of Software Architecture", *IEEE Software*, pp. 42-50, 1995.

[15] A. P. Moore, R. J. Ellison, and R. C. Linger, "Attack Modelling for Information Security and Survivability", Software Engineering Institute, 2001.

[16] OMG, "UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms", 2004.

[17] B. Schneier, "Attack Trees: Modeling Security Threats", *Dr. Dobb's Journal*, 1999.

[18] G. Sindre and A. L. Opdahl, "Eliciting Security Requirements with Misuse Cases", Proc. TOOLS-37'00, Sydney, Australia, 2000.

[19] A. Toval, J. Nicolás, B. Moros, and F. García, "Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach", *Requirements Engineering Journal*, vol. 6, pp. 205-219, 2001.

[20] VeriSign, Microsoft, SonicSoftware, IBM, BEA, and SAP, "Web Services Policy Framework (WS-Policy)", 2004.

[21] W3C, "Web Services Architecture", 2004.

[22] WS-I, "Basic Security Profile Version 1.0. Working Group Draft", 2004.

[23] J. Zhang, "Trustworthy Web Services: Actions for Now", *IEEE IT Pro*, 2005.