Segundo Taller de Seguridad en Ingeniería del Software y Bases de Datos (SISBD'2005)

Deusto (Bilbao)
6 de Junio de 2005

Eduardo Fernández-Medina y Mario Piattini (Eds.)

Segundo Taller de Seguridad en Ingeniería del Software y Bases de Datos (SISBD'2005)

Deusto (Bilbao)
6 de Junio de 2005

ACTAS

Iniciativa enmarcada en las actividades de la red RETISTIC (Red temática de investigación en el campo de la Seguridad en las Tecnologías de la Información), financiada por el Ministerio de Ciencia y Tecnología (TIC-2002-12487-E)

Presentación

La Seguridad en los sistemas de información es uno de los desafíos más importantes que están asumiendo actualmente muchas de las organizaciones. A pesar de que muchas empresas han descubierto lo crítico que resulta una correcta confidencialidad, integridad y disponibilidad de su información para el éxito de sus negocios y operaciones, muy pocas han adaptado sus sistemas para mantener la información segura, evitando accesos no autorizados, previniendo intrusos, e impidiendo el descubrimiento de información confidencial.

Actualmente, existen muchos avances tecnológicos que estimulan la utilización de sistemas de información en muchos entornos de negocio. Estos sistemas utilizan grandes cantidades de datos, que son gestionados y almacenados por bases de datos y almacenes de datos. A a menudo gestionan información que es especialmente sensible, puesto que se refieren a aspectos protegidos por las leyes de protección de datos personales (creencias, datos médicos, etc.). Por tanto, la adecuada gestión de la seguridad, así como la implantación de medidas técnicas que garanticen la seguridad de estos sistemas de información y la información que éstos gestionan resulta crucial.

Este taller se centra en analizar las aportaciones que desde la ingeniería del software y las bases de datos pueden realizarse con el fin de construir sistemas de información más seguros.

Organizadores

Eduardo Fernández-Medina (Universidad de Castilla-La Mancha)
Mario Piattini (Universidad de Castilla-La Mancha)

Grupos Participantes

- Asociación de Auditores y Auditoría y Control de Sistemas y Tecnologías de la Información y las Comunicaciones (ASIA)
- Excelentísima Diputación de Ciudad Real
- Informáticos Europeos Expertos
- Universidad Carlos III
- Universidad de Castilla La Mancha
- Universidad Católica del Maule (Chile)
- Universidad Complutense de Madrid
- Universidad Politécnica de Catalunya
- Universidad de Deusto
- Universidad de Lleida
- Universidad de Málaga
- Universidad de Murcia
- Universidad Rey Juan Carlos

Índice

Requisitos de seguridad de Servicios Web en el marco SIREN

Carlos Gutiérrez[1], Begoña Moros[3], Eduardo Fernández-Medina[2], Ambrosio Toval[3] y Mario Piattini[2]

(1) STL, Madrid(SPAIN).
carlos.gutierrez@stl.es
(2) Grupo de Investigación Alarcos. Universidad de Castilla-La Mancha.
Paseo de la Universidad 4, 13071, Ciudad Real. (España). Tel: 34 926 29 53 00
{Eduardo.FdezMedina, Mario.Piattini}@uclm.es
(3) Grupo de Investigación de Ingeniería del Software. Departamento de Informática, Lenguajes y Sistemas. Universidad de Murcia. Campus de Espinardo. 30071. Murcia (España).
{atoval, bmoros}@dif.um.es

Abstract. La seguridad en sistemas basados en servicios web es un aspecto crítico ya que basan su infraestructura operativa en Internet, que es un medio público y por tanto inherentemente, inseguro. En la actualidad existe un movimiento destacado por parte de la industria hacia la estandarización de los mecanismos de seguridad a ser utilizados en sistemas basados en servicios web. Pese a su cantidad, estos mecanismos están muy bien organizados en estándares bien clasificados. El problema está en, que dado el alto número de estos estándares y mecanismos de seguridad resulta bastante complejo saber qué mecanismos son los más adecuados y por tanto qué estándares deben ser utilizados en la práctica. Esta tarea se simplificaría enormemente si los desarrolladores dispusieran de un catálogo de plantillas de requisitos de seguridad para servicios web que pudieran reutilizar y que abarcaran el conjunto de posibles factores de la seguridad a considerar en estos sistemas. Determinados los requisitos de seguridad, el paso para identificar qué mecanismos y qué estándares de seguridad deben ser utilizados sería mucho más directo y sencillo. Este artículo presenta un catálogo de plantillas de requisitos de seguridad para servicios web basado en el proceso de reutilización de requisitos SIREN que trata de rellenar este hueco existente en el ciclo de vida de desarrollo de sistemas basados en el paradigma en cuestión.

1. Introducción

Durante los últimos años, las tecnologías de basadas en servicios web (WS) han logrado alcanzar una gran popularidad y, dicho paradigma, se encuentra entre las palabras más pronunciadas entre los gurús de la integración de los sistemas de información y expertos en tecnologías middleware. Debido a su creciente popularidad [1], los WS está acelerando el aspecto conocido como 'Computación de Confianza', que determina el éxito o fracaso de un producto software en el mercado [2].

Los WS se basan en estándares de Internet, red sobre la cual, tal y como demues' las estadísticas presentadas por el CERT, el número de incidentes relacionados con la seguridad en Internet ha crecido de manera exponencial durante los últimos años (se ha pasado de 2573 incidentes reportados en 1996 hasta los 137529 en el año 2003) [3]. Uno de los factores principales causante de este fenómeno es la incorporación de forma generalizada de las nuevas tecnologías, como aquellas basadas en Internet, en la sociedad moderna y a que las aplicaciones software son cada vez más ubicuas, heterogéneas y críticas en sus objetivos y vulnerabilidades [4].

Un aspecto determinante para la adopción definitiva por parte de la industria de las tecnologías basadas en WS es el de la seguridad. Los WS se ejecutan sobre Internet lo cual implica un riesgo inherente. Además, el trabajo llevado a cabo por los principales consorcios de la industria dirige su esfuerzo a alcanzar la completa estandarización de los mecanismos de seguridad que pueden ser aplicados en este tipo de sistemas. Esto ha provocado que existan una volumen desproporcionado de estándares de seguridad que dificultan la tarea a los desarrolladores, muchas veces no expertos en seguridad, a la hora de aplicar los mecanismos que describen. Es necesario pues, un paso previo, que dote a los desarrolladores de la capacidad de identificar fácilmente qué requisitos de seguridad son necesarios considerar en su sistemas de forma que puedan saber qué mecanismos de seguridad necesitan y, por tanto, qué estándar o estándares deben conocer con mayor profundidad. El proceso PWSSec (Proceso de Desarrollo de Seguridad en Servicios Web) [5] desarrollado por parte de los autores de este artículo presenta una aproximación guiada al desarrollo de sistemas basados en WS que integren el aspecto de la seguridad desde sus etapas más iniciales. Como parte de este proceso se describe la etapa WSSecReq (Requisitos de Seguridad en Servicios Web) que facilita la tarea de producción de una especificación de requisitos de seguridad para sistemas basados en WS. Esta etapa se basa en el proceso de reutilización de requisitos SIREN (SImple REuse of software RequiremNts) para producir una estructura de documentos de especificación de requisitos completa a partir de una jerarquía de plantillas y para elaborar un repositorio de requisitos de seguridad de WS que sea reutilizable entre proyectos.

En este artículo presentamos el resultado de aplicar SIREN en la etapa WSSecReq del proceso PWSSec mostrando, como principal producto, un catálogo de requisitos de seguridad de WS reutilizables.

Este artículo se organiza de la siguiente manera: en la sección 2 se presenta una información que sirve como base para comprender el resto del artículo; en la sección 3 se presenta el catálogo de requisitos de seguridad para WS y se incluyen una serie de plantillas de requisitos de seguridad distribuidas por el tipo de requisito de seguridad para WS al que pertenecen; en las secciones 4 y 5 se presentan las líneas de trabajo futuras así como las conclusiones, respectivamente.

2. Background

En esta sección estableceremos las bases conceptuales que permitan entender mejor el resto del artículo. En primer lugar se explica de forma breve el proceso PWSSec y su

etapa WSSecReq y, a continuación, se describe concisamente el proceso de especificación de requisitos reutilizables SIREN.

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 [6] y la integración gradual de la seguridad en los sistemas basados en WS [7]; 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 [8]: 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.

PWSSec define tres etapas de desarrollo que permiten obtener una especificación de los requisitos de seguridad (WSSecReq), posteriormente una arquitectura de seguridad (WSSecArch – Arquitectura de Seguridad para Servicios Web) y, finalmente, un diseño a bajo nivel basado en estándares de seguridad para WS (Tecnologías de Seguridad para Servicios Web). Es en la primera de estas etapas, WSSecReq, en la que se utiliza de manera directa SIREN.

SIREN [9] es un modelo de proceso de reutilización de requisitos de propósito general que puede ser aplicado en una amplia variedad de dominios y campos, como por ejemplo la seguridad, bases de datos, comercio electrónico y que incorpora de manera sistemática los aspectos de la seguridad desde las primeras etapas de desarrollo de los sistemas de información. En [10] se presenta en detalle como se aplica al dominio de la Protección de Datos Personales. Este proceso se basa en los dos siguientes elementos:

- Catálogo de requisitos que es reutilizable en cualquier proyecto software. Este catálogo se compone de una jerarquía de documentos de especificación de requisitos conformes con los estándares de especificación de requisitos de la IEEE.
- Los requisitos específicos relacionados con el proyecto sobre el que estemos trabajando.

A partir de esta información inicial, el Ingeniero de Requisitos desarrollará una especificación de requisitos con la misma estructura de plantillas que el Catálogo de Requisitos Reutilizables. Básicamente, la especificación del proyecto consistirá de los requisitos específicos directamente importados del proyecto y de los requisitos que procedan del catálogo. Estos últimos se utilizan de dos maneras: instanciando un requisito genérico del catálogo de requisitos de acuerdo a las necesidades del proyecto o incorporándolo directamente del proyecto. SIREN trata sobre cómo conseguir que el trabajo realizado en la etapa de especificación de requisitos sea reutilizable y sobre cómo reutilizarlo en otras fases de desarrollo.

3. Catálogo de Requisitos

En esta sección presentamos de forma resumida el conjunto de subfactores de la seguridad [11] que componen el catálogo de requisitos de seguridad para WS que se está desarrollando. Este catálogo es una entidad viva que se ve sujeta a una constante evolución. Para cada subfactor de seguridad se explica su interpretación en el contexto de la seguridad en WS y se muestran, a modo de ejemplo, algunas plantillas de requisitos de seguridad reutilizables. Por motivos de espacio, en este artículo sólo mostramos las plantillas de requisitos de los siguientes subfactores de la seguridad: identificación, autenticación, autorización, confidencialidad, integridad y privacidad. Pero en el catálogo se están considerando más subfactores de seguridad como: no repudio, disponibilidad, auditoria de seguridad, seguridad del perímetro, gestión de la confianza, federación y fiabilidad de la mensajería.

3.1 Fuentes

Las fuentes utilizadas para desarrollar este catálogo de requisitos han sido las siguientes:

- Mecanismos definidos en las especificaciones de seguridad de WS. Se ha hecho un estudio en profundidad del conjunto de estándares de WS (muchos de ellos aún en estado borrador) y, a partir de los mecanismos de seguridad que definen, se han extraído requisitos de seguridad de forma que se establece un relación intrínseca entre los requisitos de seguridad y los mecanismos de seguridad definidos por los estándares (destacar que en muchas ocasiones un requisito de seguridad puede ser cubierto por uno o más mecanismos de seguridad de uno o más estándares).
- Estudio de las líneas de investigación abiertas por la comunidad científica en lo relativo a la seguridad en WS.
- Proceso de elicitación de requisitos definidos por la etapa WSSecReq del proceso PWSSec creado por los autores [5, 12]. El proceso PWSSec define una etapa para la elicitación de requisitos de seguridad en sistemas basados en WS. Esta etapa utiliza SIREN con varios propósitos:
  - Crear una organización jerárquica de las distintas especificaciones de los requisitos de seguridad.
  - Crear un repositorio de plantillas de requisitos de seguridad para WS que sean reutilizables.
  - Establecer relaciones de inclusión/exclusión entre los requisitos de seguridad desarrollados.
- Se ha aplicado la etapa WSSecReq definida en el proceso PWSSec (lo que implica aplicar SIREN) sobre diferentes patrones de negocio y de aplicación para WS [12] y se han obtenidos plantillas de requisitos de seguridad reutilizables.

3.2 Plantillas

El catálogo de requisitos que aquí se presenta se basa en plantillas reutilizables entre diferentes proyectos software basados en WS. Cada parámetro puede tener uno o más valores separados por el símbolo '|' y pueden ocurrir cero o muchas veces, entonces se utiliza el símbolo '*' o una o muchas veces, entonces se utiliza el símbolo '+'.

Los parámetros básicos incluidos en las plantillas son los siguientes:

- Tipo de agente de WS: proveedor de servicios, consumidor de servicios o servicio de descubrimiento.
- Nombre de agente: que exprese el nombre del agente (ej: el nombre del servicio).
- Tipo de interacción: en WS los tipos genéricos de interacción son publicación, descubrimiento, vinculación e invocación.
- Nombre de la interacción/mensaje: nombre que identifique la interacción (ej: el nombre del caso de uso, de la actividad del flujo de trabajo o de la operación del puerto que está siendo invocada).
- Tipo de mensaje: parámetro que permite indicar el nombre del tipo de mensaje sobre el que aplica el requisito.
- Tipo de métrica: tipo de métrica utilizada para cuantificar, y poder posteriormente verificar, el requisito de seguridad.
- Valor de la métrica: valor asignado al tipo de métrica utilizado para cuantificar el criterio de seguridad seleccionado.
- Tipo de ataque: cuyo valor puede ser 'no sofisticado', 'semisofisticado' o 'sofisticado'.
- Tipo de atacante: cuyo valor depende de los tipos de atacantes identificados en el sistema.
- Nivel de protección de un mensaje: cuyo valor puede ser 'a nivel de la capa de transporte' o 'a nivel de la capa de mensaje'.

3.2.1 Identificación

La identificación en el contexto de los WS implica que cierta interacción entre WS consumidores y proveedores requiera que en los mensajes intercambiados existan elementos que los identifiquen (a ellos directamente o a las entidades a favor de quién realizan las peticiones). Un ejemplo de plantilla de requisito de seguridad para WS definido de este tipo es:

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] requerirá al [[agente consumidor | agente proveedor | agente descubrimiento] [nombre agente]]+ que se identifique [a nivel de servicio | en nombre de otra entidad | ambos] a nivel de la capa de [transporte <protocolo>| mensaje SOAP |ambos] durante la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ con cierta [métrica] "

Vemos como en esta plantilla de se incluye la semántica de que un agente proveedor de WS requiera a sus agentes consumidores que se identifiquen ellos mismos o en nombre de un tercero.

3.2.2 Autenticación

La autenticación en sistemas distribuidos se puede dividir en dos procesos:

- Autenticación de las entidades que consiste en verificar la identidad declarada por un agente de WS. Esta identidad puede ser la identidad del propio agente o la identidad de una tercera persona a favor de quién el agente está realizando la petición (o ambas).
- Autenticación de los mensajes que consiste en saber con certeza que cierto mensaje es genuino: procede de la fuente que lo generó y sin modificaciones alguna.. La autenticación de los mensajes es un procedimiento que permite la comunicación entre las partes para verificar que los mensajes recibidos son auténticos. Se dice que un mensaje (ej: SOAP) es auténtico cuando es genuino y procede de la fuente original [13]. La autenticación de los mensajes, también conocida en la literatura como autenticación del origen de los datos, es un tipo de autenticación mediante el cual se garantiza que una parte es la fuente original de los datos creados en algún instante anterior (normalmente no definido) [14]. "Por definición, la autenticación del origen de los datos incluye integridad de los datos" [14].

A continuación se muestran dos ejemplos de plantillas de requisitos de seguridad de autenticación de entidades:

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] verificará la identidad proporcionada por el [[agente consumidor | agente proveedor | agente descubrimiento] [nombre agente]]+ a nivel de la capa de [transporte <protocolo>| mensaje SOAP |ambos] con el objetivo de evitar ataques [no sofisticados | semisofisticados | sofisticados] durante la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ con cierta [métrica] ".

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] [detectará | prevendrá | ambos] la reutilización de la información de autenticación a nivel de la capa de [transporte <protocolo>| mensaje SOAP |ambos] con el objetivo de evitar ataques [no sofisticados | semisofisticados | sofisticados] de repetición durante la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ con cierta [métrica] ".

Y a continuación se muestra otra plantilla relacionada con la autenticación de los mensajes:

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] verificará la autenticidad del mensaje(s) [nombre]+ enviado por el [[agente consumidor | agente proveedor | agente descubrimiento] [nombre agente]]+ a nivel de la capa de [transporte <protocolo> | mensaje SOAP | ambos] con el objetivo de evitar ataques [no sofisticados | semisofisticados | sofisticados] durante la ejecución del [tipo de interacción] [interacción | caso de uso] con cierta [métrica] ".

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] detectará que el/los mensaje(s) [nombre]+ enviado por el [[agente consumidor | agente proveedor | agente descubrimiento] [nombre agente]]+ NO son auténticos a nivel de la capa de [transporte <protocolo> | mensaje SOAP | ambos] durante la manifestación de ataques [no sofisticados | semisofisticados | sofisticados] durante la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ con cierta [métrica] ".

3.2.3 Autorización

En el contexto de los WS la autorización consiste básicamente en garantizar que sólo los agentes consumidores de WS autorizados, y bajo las circunstancias apropiadas, pueden acceder a los servicios ofrecidos por cierto agente proveedor de WS. El estándar de referencia en este sentido es XACML (eXtensible Access Control Mark-up Language). A partir de los mecanismos definidos en este estándar y de aquellos definidos en soluciones propuestas como [15-18] hemos elaborado un conjunto de plantillas de requisitos de seguridad relativos a la autorización como la siguiente:

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] no autorizará a [[agente consumidor | agente proveedor | agente descubrimiento] [nombre agente]]+ que no dispongan de los privilegios adecuados a nivel de la capa de [transporte <protocolo> | mensaje SOAP | ambos], con el objetivo de evitar ataques [no sofisticados | semisofisticados | sofisticados], la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ con cierta [métrica] ".

3.2.4 Confidencialidad

La confidencialidad en WS se considera principalmente en el contenido de los mensajes intercambiados. La manera más común de garantizar la confidencialidad de los mensajes es utilizando criptografía.

Un ejemplo de plantilla de requisito del catálogo elaborado es:
"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] deberá proteger el mensaje [nombre del mensaje] a nivel de la capa de [transporte <protocolo>| mensaje SOAP |ambos] que transmite de forma que los [datos críticos] sólo sean visibles al [[agente consumidor | agente proveedor | agente descubrimiento] [nombre agente]]*, resistiendo ataques [no sofisticados | semisofisticados | sofisticados] durante la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ con cierta [métrica]".

3.2.5 Integridad

La integridad en el contexto de los WS está principalmente asociada con la integridad de los mensajes que se intercambian. Además, específico para este tipo de sistemas está el concepto de integridad del comportamiento de un WS que representa que cierto agente proveedor de WS se comportará tal cual está descrito en su política [19].

Dada la arquitectura de los WS, en la que existe el rol de nodo SOAP intermediario con capacidad potencial de procesar los mensajes enviados entre un emisor y un destinatario final, la integridad en los mensaje SOAP, y de su contenido XML, está basada en su semántica. Es decir, cierto mensaje recibido por un receptor final puede no ser idéntico al original enviado por el emisor, por ejemplo porque fue eliminada cierta cabecera por un nodo SOAP intermediario, y sin embargo, si su semántica está intacta seguir siendo aún válido.

Un ejemplo de plantilla de requisito de seguridad concerniente con la integridad de los mensajes es la siguiente:

"El [agente consumidor | agente proveedor | agente descubrimiento] [nombre agente] deberá proteger el mensaje [nombre del mensaje] a nivel de la capa de [transporte <protocolo>| mensaje SOAP |ambos] que transmite de posibles [modificaciones | eliminaciones | inserciones] sobre [partes del mensaje] que alteren su semántica debido a ataques [no sofisticados | semisofisticados | sofisticados] durante la ejecución del [[tipo de interacción] [interacción | caso de uso]]+ ".

3.2.6 Privacidad

El principal propósito de la privacidad en WS consiste en garantizar la custodia correcta (es decir la correcta divulgación) de la información crítica o sensible almacenada por cierta entidad u organización acerca de sus clientes. La infraestructura completa sobre la privacidad ya ha sido establecida por el consorcio W3C con la publicación de las recomendaciones P3P (Policy for Privacy Preferences) [20], APPEL (P3P Preference Exchange Language) [21] y E-P3P o EPAL [22]. Hemos estudiado estas recomendaciones para analizar los mecanismos propuestos y derivar así las plantillas de requisitos de seguridad reutilizables.

Otro aspecto a considerar en lo referente a la privacidad de la información guarda relación con los WS de descubrimiento ya que estos son los responsables de almacenar la información relacionada con las organizaciones y sus servicios que pertenece a las organizaciones proveedoras de WS. Por tanto, aquí la privacidad de los datos se antoja crítica de forma que sólo los clientes autorizados deberán poder publicar, modificar, acceder y eliminar la información almacenada en estos servicios de descubrimiento.

Otro aspecto de privacidad adicional que cobra gran importancia en escenarios de federación guarda relación con la federación de los atributos de los principales pertenecientes a cada uno de los dominios de confianza. Cuando se federa la autorización, los Puntos de Decisión de Políticas (PDP) residentes en los dominios de confianza de los WS con los que se desea interaccionar suelen necesitar información, en forma de atributos, de los principales que intentan acceder. Estos atributos de confianza suelen residir en un almacén de atributos cuyo acceso es intermediado por un servicio de atributos. Cuando el PDP del agente proveedor de WS solicita información sobre ciertos atributos del cliente, el servicio de atributos perteneciente al dominio de confianza del cliente debe aplicar una política de privacidad adecuada que evite la divulgación del valor de estos atributos a los PDP no autorizados (ej: PDPs de dominios de confianza que no pertenecen a la federación).

El anonimato, como tipo de requisito de privacidad [23], tiene especial relevancia en entornos de federación en los debe resultar imposible, si es que fuera requisito, reconstruir la traza de las operaciones ejecutadas por cierto usuario sobre servicios desplegados en ciertos dominios de confianza distintos al suyo. Estos escenarios de federación se habilitan gracias a las interacciones llevadas a cabo por WS residentes en cada uno de los dominios de confianza federados y que son transparentes para el usuario que transita por ellos.

Además de todos estos aspectos, la privacidad tiene connotaciones legales que se deben tener en cuenta a la hora de determinar las políticas de privacidad que serán aplicadas con los WS desarrollados. Si las políticas de privacidad aplicadas por los WS no se ajustan a las regulaciones de privacidad legales, la organización proveedora de los WS puede estar incurriendo en graves delitos legales [10].

Un ejemplo de plantilla de requisito de seguridad relativa a la privacidad de la información sensible, perteneciente a ciertos sujeto vinculado con cierto dominio de confianza, y que cuya información es consultada por servicios (ej: Puntos de Decisión de Políticas) pertenecientes a otros dominios podría ser el siguiente:

"El [agente consumidor | agente proveedor | agente descubrimiento] garantizará la no divulgación de su información [[tipo] [identificador]]+ sin el consentimiento expreso de su propietario al [agente consumidor | agente proveedor | agente descubrimiento] durante la ejecución de [[tipo de interacción] [interacción | caso de uso]]+ según el criterio y medidas dado en la tabla [tabla]".

4. Trabajo Futuro

La línea de investigación principal abierta en la actualidad consiste en completar y consolidar el catálogo de plantillas de requisitos de seguridad definidos. Actualmente se está completando realizando una revisión en profundidad y detallada de las fuentes que se han utilizado para rellenar el catálogo. Además se está tratando de consolidar el catálogo eliminado redundancias, ambigüedades y conflictos entre los requisitos y tratando de refinar la organización de los mismos.

Además, actualmente estamos analizando, como parte de la aplicación del proceso SIREN, las posibles relaciones de inclusión y exclusión que existente entre estas plantillas de requisitos.

Además, queremos incorporar estas plantillas, y sus meta-atributos tal cual definidos en SIREN (identificación, prioridad, criticidad, etc.) en una herramienta CASE que dé soporte a la etapa de requisitos WSSecReq, como por ejemplo Rational Requisite Pro, de forma que ésta se vea altamente asistida para el desarrollador.

5. Conclusiones

En este artículo hemos presentado cómo se ha utilizado el proceso de reutilización de requisitos SIREN para generar un catálogo de requisitos de seguridad de WS que sean reutilizables. La aplicación de SIREN se ha realizado en el contexto del proceso de desarrollo PWSSec y, más concretamente, en su etapa dedicada a la especificación de requisitos de seguridad, WSSecReq.

El resultado de esta interacción entre SIREN y WSSecReq ha sido la creación de un catálogo de requisitos de seguridad reutilizables (la mayor parte de estos requisitos están parametrizados, es decir, son plantillas) para sistemas basados en WS.

Agradecimientos

Este artículo ha sido realizado en el marco del proyecto CALIPO (TIC2003-07804-CO5-03) y la red RETISTIC (TIC2002-12487-E), de la Dirección General de Investigación del Ministerio de Ciencia y Tecnología.

Referencias

1. Zhang, J., *Trustworthy Web Services: Actions for Now.* IEEE IT Pro, 2005.
2. Tian, J., *Quality-Evaluation Models and Measurements.* IEEE Software, 2004: p. 84-91.
3. CERT.
4. Devanbu, P.T. and S. Stubblebine. *Software Engineering for Security: a Roadmap.* in *The Future of Software Engineering, Special Volume published in conjuction with ICSE.* 2000. Limerick, Ireland.
5. Gutiérrez, C., E. Fernández-Medina, and M. Piattini. *PWSSec: Process for Web Services Security.* in *IEEE International Conference on Web Services 2005.* 2005. Orlando, Florida, USA.
6. Boehm, B.W., *A Spiral Model of Software Development and Enhancement.* IEEE Computer, 1988: p. 61-72.
7. Breu, R., et al. *Key Issues of a Formally Based Process Model for Security Engineering.* in *ICSSEA03.* 2003.
8. W3C, *Web Services Architecture.* 2004.
9. Toval, A., et al., *Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach.* Requirements Engineering Journal, 2001. 6(4): p. 205-219.
10. Toval, A., A. Olmos, and M. Piattini. *Legal Requirements Reuse: A Critical Success Factor for Requirements Quality and Personal Data Protection.* in *IEEE Joint International Conference on Requirements Engineering (RE'02).* 2002. University of Essen, Germany.
11. Firesmith, D.G., *Specifying Reusable Security Requirements.* Journal of Object Technology, 2004. 3(1): p. 61-75.
12. Gutiérrez, C., E. Fernández-Medina, and M. Piattini. *Desarrollo de sistemas de servicios web seguros.* in *JSWEB'05.* 2005. Granada, Spain.
13. Stallings, W., *Network Security Essentials.* 2 ed. 2003: Prentice Hall. 409.
14. Menezes, A., P.v. Oorschot, and S. Vanstone, *Chapter 9. Hash functions and Data Integrity. 9.6. Data integrity and message authentication*, in *Handbook of Applied Cryptography.* 1996, CRC Press.
15. Damiani, E., et al. *Fine grained access control for SOAP e-services.* in *Proc. of 10th Int. Conf.on World Wide Web (WWW).* 2001.
16. Koshutanski, H. and F. Massacci. *An Access Control Framework for Business Processes for Web Services.* in *ACM Workshop on XML Security.* 2003.
17. Bertino, E., et al. *Specifying and enforcing access control policies for XML document sources.* in *World Wide Web.* 2000.
18. Wonohoesodo, R. and Z. Tari. *A Role based Access Control for Web Services.* in *ICWS'04.* 2004. San Diego, California, USA.
19. Zhang, L.-J., J. Zhang, and J.-Y. Chung. *An Approach to Help Select Trustworthy Web Services.* in *E-Commerce Technology for Dynamic Business, IEEE International Conference on CEC-East'04.* 2004. Beijing, China.
20. Cranor, L., et al., *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification.* 2002.
21. Cranor, L., M. Langheinrich, and M. Marchiori, *A P3P Preference Exchange Language 1.0 (APPEL1.0).* 2002.
22. Ashley, P., et al. *E-P3P Privacy Policies and Privacy Authorization.* in *WPES'02.* 2002. Washington, DC, USA: ACM.
23. Firesmith, D.G., *Common Concepts Underlying Safety, Security, and Survivability Engineering.* 2003, SEI.