tecnim@p Sevilla 2006
30 mayo a 2 junio
060.es
24x7x365

Comunicación

048

COMPLEMENTANDO A MÉTRICA: PROCESO DE INGENIERÍA DE REQUISITOS DE SEGURIDAD PARA EL DESARROLLO DE SISTEMAS DE INFORMACIÓN SEGUROS

Daniel Mellado Fernández
Técnico Superior de Informática
Gerencia de Informática de la Seguridad Social

Eduardo Fernández-Medina Patón
Profesor Asociado a tiempo completo
Universidad de Castilla-La Mancha

Mario Piattini Velthuis
Catedrático de Universidad
Universidad de Castilla-La Mancha

Daniel Mellado Fernández
Eduardo Fernández-Medina Patón
Mario Piattini Velthuis

Palabras clave

Seguridad, ingeniería de requisitos de seguridad, ingeniería de requisitos, MÉTRICA, MAGERIT

Resumen de su Comunicación

La integración de la seguridad en las primeras fases del desarrollo de Sistemas de Información (SI) es necesario si se quiere construir SI seguros. Sin embargo, en muchos proyectos software la seguridad se trata cuando el sistema ya ha sido diseñado y puesto en operación. Esta comunicación plantea una propuesta llamada SREP (Security Requirements Engineering Process – Proceso de Ingeniería de Requisitos de Seguridad) para el desarrollo de SI seguros y que viene a complementar a MÉTRICA versión 3 y a MAGERIT versión 2 en el tratamiento de requisitos de seguridad. Esta propuesta está basada en la reutilización de requisitos de seguridad, proporcionando un repositorio de recursos de seguridad, y en la integración de los Criterios Comunes en el ciclo de vida del software, siendo de esta forma conforme al estándar ISO/IEC 15408. Asimismo, SREP sigue del código de buenas prácticas de gestión de la seguridad del estándar ISO/IEC 17799:2005. Partiendo del concepto de construcción iterativa de software, proponemos un micro-proceso para el análisis de requisitos de seguridad que se realiza repetidamente en cada uno de los procesos de MÉTRICA a lo largo del desarrollo incremental. En resumen, presentamos una propuesta que permite el tratamiento sistemático e intuitivo de los requisitos de seguridad desde las primeras fases del desarrollo software.

Daniel Mellado Fernández
Eduardo Fernández-Medina Patón
Mario Piattini Velthuis

COMPLEMENTANDO A MÉTRICA: PROCESO DE INGENIERÍA DE REQUISITOS DE SEGURIDAD PARA EL DESARROLLO DE SISTEMAS DE INFORMACIÓN SEGUROS

1. Introducción

En los últimos años hemos observado como cada vez más organizaciones se hacen altamente dependientes de los Sistemas de Información (SI). Sin embargo, las aplicaciones software son cada vez más ubicuas, heterogéneas, críticas para la misión de la organización y vulnerables a incidentes de seguridad intencionados y no intencionados [3, 9]. Por lo tanto en la actual Sociedad de la Información es absolutamente vital que los SI sean asegurados apropiadamente desde el principio [1, 13] debido a las potenciales pérdidas a las que se enfrentan las organizaciones que confían en todos estos SI.

Como sabemos, es ampliamente aceptado el principio que establece que la construcción de la seguridad en las etapas tempranas del proceso de desarrollo es más eficaz respecto a los costes y tiene como resultado diseños más robustos [10]. Sin embargo, el gran problema es que en la mayoría de los proyectos software la seguridad se trata una vez el sistema ha sido diseñado e implementado [5].

Una parte muy importante en el proceso de desarrollo software para conseguir SI seguros es la denominada Ingeniería de Requisitos de Seguridad. La cual proporciona técnicas, métodos y normas para abordar esta tarea en el ciclo de desarrollo de los SI. Y debería implicar 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 del sistema [11]. Un buen documento de requisitos debe incluir tanto requisitos funcionales (relativos a los servicios que el software o sistema debe proporcionar) y los no-funcionales (relativos a las denominadas características de calidad, como rendimiento, portabilidad, seguridad etc.) [5]. Por su parte, la seguridad debe ser considerada durante todo el proceso de desarrollo y debería ser definida conjuntamente con la especificación de requisitos [17].

En esta comunicación se analizará el tratamiento de los requisitos de seguridad en MÉTRICA v.3 y su interfaz de seguridad con MAGERIT v.2, llegándose a la conclusión de que no son tratados de forma suficientemente concreta ni se proponen técnicas lo suficiente específicas para la definición de requisitos de seguridad. Por tanto, habiendo previamente realizado un análisis comparativo en [15] y [16] de varias propuestas relevantes de requisitos de seguridad en SI, como son las de Toval et al. 2001 [21], Popp et al. 2003 [19], Firesmith 2003 [7], Breu et al. 2004 [2], etc. , planteamos una nueva propuesta llamada SREP (Security Requirements Engineering Process – Proceso de Ingeniería de Requisitos de Seguridad) que viene a complementar a MÉTRICA versión 3 y a su interfaz de seguridad con MAGERIT versión 2 en el establecimiento de requisitos de seguridad. SREP describe cómo integrar los requisitos de seguridad en el proceso de ingeniería del software de una forma sistemática e intuitiva, para lo cual se basa en la integración de los Criterios Comunes (CC) en el modelo de ciclo de vida de desarrollo, ya que éstos nos ayudan con el tratamiento de los requisitos de seguridad a lo largo del ciclo de vida, asimismo SREP se apoya en la reutilización de recursos de seguridad (activos, objetivos de seguridad, amenazas y requisitos). Por último, con el fin de soportar el tratamiento de requisitos de seguridad desde las primeras fases de desarrollo, SREP plantea el uso varias técnicas y conceptos: un repositorio de recursos de seguridad, el uso de UMLSec [19], de casos de mal uso [20], de casos de uso de seguridad [7], de árboles de ataque. Facilitándose así la especificación de amenazas y requisitos de seguridad, al igual que la reutilización, de tal modo que se reutilice y se almacene como conocimiento explícito la mayor cantidad de conocimiento de seguridad y se facilite así su uso por usuarios no experimentados en la ingeniería de requisitos de seguridad (almacenándose y reutilizándose no solo requisitos, también amenazas, activos, etc.).

El resto de la comunicación está organizada de la siguiente forma: en la sección 2, se analiza el tratamiento de los requisitos de seguridad en MÉTRICA v.3 en las primeras fases; a continuación, se presenta SREP en la sección 3; y finalmente, presentamos nuestras conclusiones en la sección 4.

2. Tratamiento de los Requisitos de Seguridad en MÉTRICA v.3 en las primeras fases

MÉTRICA v.3 es una metodología que se estructura en 3 procesos principales (Planificación, Desarrollo y Mantenimiento), 5 subprocesos en los que se subdivide el proceso de desarrollo (Estudio de Viabilidad - EVS, Análisis - ASI, Diseño - DSI, Construcción - CSI, Implantación y Aceptación - IAS, y Mantenimiento - MSI) y 4 interfaces (Gestión de Proyectos, Gestión de Configuración, Aseguramiento de Calidad y Seguridad - SEG). Los procesos están formados por actividades y éstas por tareas. Para cada tarea se describe su contenido haciendo referencia a sus principales acciones, productos, técnicas, prácticas y participantes. El orden de las actividades no es secuencial y no se termina un proceso hasta no haberse realizado todas sus actividades. En una única estructura la metodología MÉTRICA v.3 cubre distintos tipos de desarrollo: estructurado y orientado a objetos, facilitándolo a través de sus 4 interfaces la realización de los procesos de apoyo u organizativos.

Los requisitos de seguridad en MÉTRICA v.3 se tratan, aunque no de manera muy detallada, en las siguientes actividades: en EVS 3 "Definición de requisitos del sistema", las tareas EVS 3.2 y 3.3 "Identificación y Catalogación de requisitos"; ASI 2 "Establecimiento de Requisitos"; en ASI 9 "Análisis de consistencia y especificación de requisitos", la tarea ASI 9.4 "Elaboración de la especificación de requisitos software"; en DSI 1 "Definición de la arquitectura del sistema" la tarea DSI 1.7 "Especificación de requisitos de operación y seguridad". En su interfaz de seguridad se tratan de manera específica en las siguientes actividades: PSI-SEG 1 "Planificación de la seguridad requerida en el proceso PSI", EVS-SEG 3 "Recomendaciones adicionales de seguridad para el sistema de información", ASI-SEG 2 "Descripción de las funciones y mecanismos de seguridad", ASI-SEG 3 "Definición de los criterios de aceptación de la seguridad", DSI-SEG 2 "Especificación de requisitos de seguridad del entorno tecnológico", DSI-SEG 3 "Requisitos de seguridad del entorno de construcción". Además las principales prácticas y técnicas utilizadas son: sesiones de trabajo, catalogación, casos de uso, revisión. Así como los principales roles que intervienen son: Jefe de proyecto, analista, equipo de seguridad, responsable de seguridad, usuario experto.

Finalmente, tras haberse expuesto y analizado someramente el tratamiento de los requisitos de seguridad en MÉTRICA v.3, llegamos a la conclusión de que MÉTRICA v.3 y su interfaz de seguridad con MAGERIT v.2 no son lo suficientemente específicas para un tratamiento sistemático e intuitivo de los requisitos de seguridad en las primeras fases del desarrollo software, ni se proponen técnicas lo suficiente específicas para la adecuada, eficaz y sencilla definición de requisitos de seguridad.

3. SREP: Proceso de Ingeniería de Requisitos de Seguridad

En esta sección describimos brevemente SREP (Security Requirements Engineering Process – Proceso de Ingeniería de Requisitos de Seguridad) que viene a complementar a MÉTRICA versión 3 y a su interfaz de seguridad con MAGERIT versión 2 en el tratamiento de requisitos de seguridad, y donde explicamos como SREP complementa a MÉTRICA en las primeras fases (procesos), quedándose fuera del alcance de esta comunicación una más extensa y precisa descripción tanto de SREP como de su acoplamiento con MÉTRICA en todos sus procesos.

Daniel Mellado Fernández
Eduardo Fernández-Medina Patón
Mario Piattini Velthuis

3.1 Visión General de SREP

SREP es un proceso basado en los activos y dirigido por el riesgo para el establecimiento de requisitos de seguridad en el desarrollo de SI seguros. Este proceso básicamente describe cómo integrar los CC en el ciclo de vida tradicional del software, junto con el uso de un repositorio de recursos de seguridad que permita la reutilización de requisitos de seguridad (los cuales pueden estar modelados con UMLSec, o expresados como casos de uso de seguridad, o como texto plano con una especificación formal), activos, amenazas (que pueden ser representadas mediante casos de mal uso, árboles de ataque, diagramas UMLSec o en texto plano por ejemplo en forma de lista de verificación ('checklist')) y medidas de salvaguarda (contramedidas). El foco de este proceso busca construir la seguridad del SI en las primeras fases del ciclo de vida de desarrollo.

Como se observa en la Fig. 1, el núcleo de SREP es un micro-proceso, que se compone de nueve actividades que son realizadas repetidamente en cada fase del ciclo de vida, pero con diferente énfasis en las actividades en función de la fase en la que se esté. Ya que el modelo elegido por SREP es iterativo e incremental y los requisitos de seguridad se van refinando a lo largo del ciclo de vida, aunque todos los principales requisitos de seguridad habrán sido identificados en la fase de requisitos del proyecto (en el análisis en MÉTRICA v.3), de acuerdo con la ISO/IEC 17799:2005. Sin embargo, por ejemplo durante el diseño se puede mejorar la especificación con requisitos relacionados con el entorno tecnológico y con las contramedidas asociadas. Al mismo tiempo, los Componentes de los CC son introducidos en el ciclo de vida del software de manera que SREP usa los diferentes Componentes CC según la fase, aunque las actividades de Aseguramiento de la Calidad (SQA) se realizan a lo largo de todas las fases del ciclo de vida de desarrollo del software. Y son en estas actividades de SQA donde los requisitos de aseguramiento de los CC deben ser incorporados, de acuerdo con S.H. Kam [8]. En las secciones siguientes se presentará una exposición un poco más detallada de la integración de los CC y de SREP en ciclo de vida de desarrollo del SI.

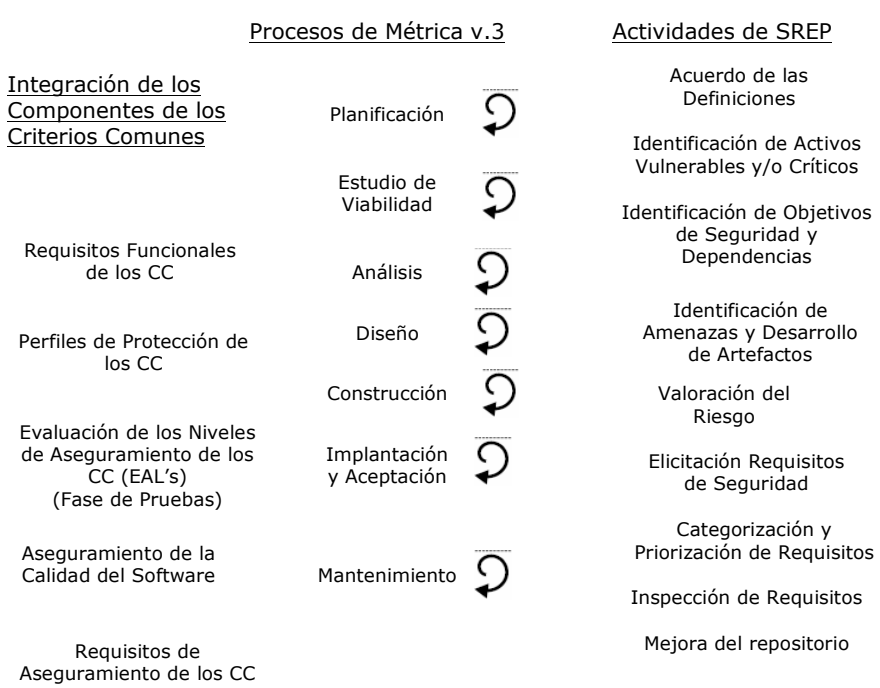

Figura 1. Visión general de SREP

Daniel Mellado Fernández
Eduardo Fernández-Medina Patón
Mario Piattini Velthuis

3.2 El Repositorio de Recursos de Seguridad

El propósito del desarrollo con reutilización de requisitos es identificar descripciones de sistemas que pueden ser usadas (ya sea total o parcialmente) con un número mínimo de modificaciones, reduciéndose así el esfuerzo total de desarrollo [4]. Además, la reutilización de requisitos de seguridad ayuda a incrementar su calidad: inconsistencias, errores, ambigüedades y otros problemas pueden ser detectados y corregidos para poder ser usados en proyectos sucesivos [21]. De esta manera, se nos garantizará que los ciclos de desarrollo son realizados lo más rápido posible y estando a la vez basados en soluciones probadas. Por ello, proponemos un Repositorio de Recursos de Seguridad (RRS).

El RRS almacena todos los elementos reutilizables que pueden ser usados por los analistas o ingenieros de requisitos. Además, el repositorio entiende los conceptos de dominio y perfil, de forma similar al método SIREN [21]. El primero consiste en la pertenencia a un campo específico o a unas áreas funcionales de aplicación, como el comercio electrónico. En cambio, el concepto de perfil se refiere a un conjunto homogéneo de requisitos que pueden ser aplicados a diferentes dominios, como por ejemplo la legislación relativa a la protección de datos de carácter personal. En relación con esto, nosotros planteamos implementar los dominios y perfiles aprovechando los conceptos de 'Paquetes' y 'Perfiles de Protección' (PP) de los CC. De esta manera, los requisitos son guardados como subconjuntos estandarizados de requisitos de seguridad específicos, y con ellos sus elementos asociados en el RRS (amenazas, etc.). En resumen, cada dominio o perfil son una vista del RRS global. Adicionalmente, los elementos que el RRS contiene son establecidos de forma genérica, mediante el uso de mecanismos basados en parámetros, como plantillas parametrizadas reutilizables. Aunque hay también plantillas no parametrizadas y listas de verificación ('checklists').

A continuación se muestra en la Fig. 2 el meta-modelo del RRS, el cual es una extensión (representada mediante fondo oscuro) de la propuesta del repositorio formulada por Sindre, G., D.G. Firesmith, y A.L. Opdahl [20].

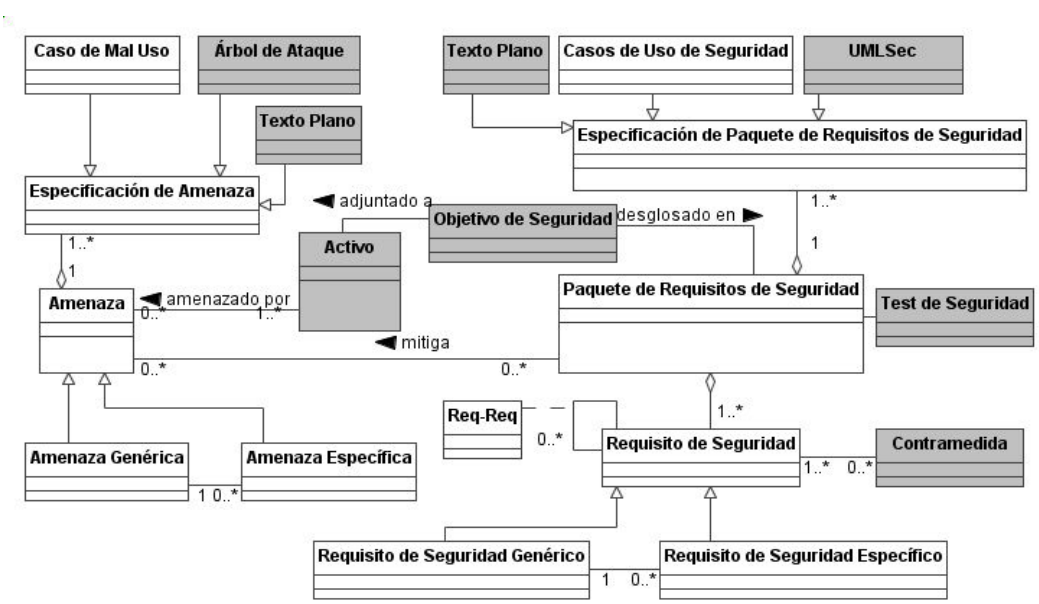

Figura 2. Meta-modelo del repositorio de recursos de seguridad

Tal y como se presenta anteriormente, se trata de un meta-modelo dirigido por los activos y por las amenazas, porque los requisitos pueden obtenerse a través de los activos o de las amenazas. Seguidamente, describimos brevemente los más importantes y/o complejos aspectos del meta-modelo.

- 'Amenaza Genérica' y 'Requisito de Seguridad Genérico' describen amenazas y requisitos independientemente de los dominios particulares. Y pueden ser representados con especificaciones distintas gracias a los elementos 'Especificación de Amenaza' y 'Especificación de Paquete de Requisitos de Seguridad'.
- 'Paquete de Requisitos de Seguridad' es un conjunto de requisitos que se juntan para satisfacer los mismos objetivos de seguridad y mitigar las mismas amenazas. Ya que estamos de acuerdo con Sindre, G., D.G. Firesmith, y A.L. Opdahl [20] en que en muchas ocasiones son una mayor y más efectiva unidad de reutilización.
- La relación 'Req-Req' permite establecer relaciones de inclusividad o exclusividad entre requisitos. Una relación de exclusividad entre requisitos implica que son alternativos mutuamente, como por ejemplo si están en conflicto o se solapan uno al otro. Mientras que una relación de inclusividad entre requisitos significa que para satisfacer un requisito dado, se necesita satisfacer también otro u otros.

Además, pueden producirse más relaciones más adelante en el momento de la especificación del diseño, de los casos de prueba de seguridad, medidas de salvaguarda, etc. Debido a que nuestro modelo de proceso propuesto esta basado en el concepto de la construcción iterativa de software, como se explica en la siguiente sección.

Por último, queremos destacar el hecho de que usando los CC, un gran número de requisitos de seguridad pueden ser definidos en el propio sistema y en el desarrollo del mismo. Sin embargo, los CC no proporcionan soporte metodológico, ni contienen criterios de evaluación de la seguridad relativos a las medidas de seguridad administrativas no directamente relacionadas con las medidas de seguridad del SI. Aunque se sabe que una gran parte de la seguridad que alcanza un SI se obtiene mediante la aplicación de estas medidas administrativas. Por tanto, y de acuerdo con la norma ISO/IEC 17799:2005, proponemos incluir el conjunto de requisitos legales, estatutarios, regulatorios, y contractuales que deberían satisfacer la Organización, sus socios comerciales, los contratistas, y los proveedores de servicios, y su entorno socio-cultural. Con lo que después de adecuar estos requisitos al formato de los requisitos del sistema o bien requisitos software, éstos constituirán el subconjunto inicial de requisitos de seguridad del RRS para cualquier proyecto. Además, si la Organización realiza alguna actividad en España de manera directa o indirecta, nosotros planteamos que el RRS contenga todos los elementos del catálogo (activos, amenazas, salvaguardas,...) de MAGERIT v.2, que es también conforme a la ISO/IEC 15408, y siguiéndose la notación propuesta en dicho catálogo. De forma que constituya así un perfil mediante el cual se facilite la conformidad con la legislación española relativa a la seguridad y la protección de datos de carácter personal.

3.3. Modelo de Proceso de SREP

Partiendo del concepto de construcción iterativa de software, planteamos un micro-proceso que se compone de nueve actividades que son realizadas repetidamente en cada fase del ciclo de vida, pero con diferente énfasis en las actividades en función de la fase en la que se esté, y en cada iteración se generarán versiones internas o externas de varios artefactos, de forma que todos juntos constituyan una línea base. De esta manera el documento de "Especificación de Requisitos de Seguridad" evolucionará a lo largo del ciclo de vida. Además, cada requisito de seguridad puede ser localizado a través de los distintos niveles de abstracción, y también, como el modelo entiende el concepto de dominio, éstos podrán ser analizados por las partes interesadas ("stakeholders") que tengan más conocimiento y/o responsabilidad en cada dominio. Asimismo, estamos de acuerdo con Nuseibeh [18] en que los procesos de Ingeniería de Requisitos y el Diseño de la Arquitectura son procesos concurrentes y que se influencian mutuamente.

Las nueve actividades (basadas en [14] y [20]) que constituyen el micro-proceso para el análisis de requisitos de seguridad, junto con los artefactos visibles y externos que son generados en estas actividades, se presentan a continuación:

- Actividad 1: Acuerdo de las definiciones. La primera tarea para la Organización es definir las partes interesadas ("stakeholders") y acordar un conjunto común de definiciones de seguridad, junto con la definición de las políticas de seguridad de la Organización y la visión de seguridad del sistema. De manera que en esta actividad es cuando se genera el 'Documento de la Visión de la Seguridad del SI'. Además, las partes interesadas participarán en estas últimas tareas y las definiciones candidatas serán principalmente extraídas de los estándares de IEEE e ISO/IEC, como los siguientes: ISO/IEC 13335, ISO/IEC 17799:2005, ISO/IEC 15408, ISO/IEC 21827, IEEE 830:1998, IEEE 1061-1992.

- Actividad 2: Identificación de activos vulnerables y/o críticos. Es en este punto donde el RRS puede ser usado por primera vez. Y consiste en la identificación de los diferentes tipos de activos valiosos o críticos y/o vulnerables, y es realizado por el ingeniero de requisitos, que puede ayudarse mediante:

- Listas de activos extraídas del RRS, donde los activos pueden buscarse por dominio, e incluso puede ser seleccionado un perfil concreto.
- Los requisitos funcionales.
- Las entrevistas con las partes interesadas ('stakeholders').

- Actividad 3: Identificación de objetivos de seguridad y dependencias. En esta actividad el RRS puede ser también usado, ya que cada activo tiene adjuntado unos objetivos de seguridad. Para cada activo identificado en el paso anterior se seleccionan los requisitos de seguridad apropiados para el activo, y se identifican las dependencias entre ellos. Asimismo en esta actividad es cuando los objetivos de seguridad para el entorno son identificados y son hechas las suposiciones acerca del mismo. Los objetivos de seguridad se expresan especificando el nivel de seguridad necesario en términos de probabilidad y en tipos probables de atacantes. Y quedarán plasmados en esta actividad en el 'Documento de Objetivos de Seguridad', el cual podrá ser refinado en sucesivas iteraciones (en los procesos de MÉTRICA de planificación y desarrollo, quedándose totalmente definidos al acabar el análisis).

- Actividad 4: Identificación de amenazas y desarrollo de artefactos. Cada activo es amenazado por una/s amenaza/s que pueden impedir la consecución del objetivo/s de seguridad de dicho activo. Con lo que primeramente hay que encontrar todas las amenazas que apuntan sobre los activos, con la ayuda del RRS. Además, puede que sea necesario desarrollar artefactos (como casos de mal uso, árboles de ataque, o casos de uso y diagramas de clases o de secuencia o estado usando UMLSec) para especificar nuevas amenazas y/o requisitos genéricos y específicos. Ya que es necesario buscar por nuevas amenazas que no estén relacionadas con los activos a través del RRS, porque es posible que algunos objetivos de seguridad y activos se hayan podido olvidar en pasos anteriores o bien dichas amenazas no hayan sido introducidas aún en el RRS. Llegado este punto puede que sea posible utilizar algún o algunos Perfiles de Protección (PP) o Paquetes de los CC o de sus adaptaciones existentes en nuestro repositorio y adaptarlos para cumplir con los requisitos del SI. Finalmente, se genera el 'Documento de Definición del Problema de Seguridad', que contendrá la definición del problema de seguridad existente, las amenazas, suposiciones y afirmaciones de conformidad (con PP's, etc.), y el cual podrá ser refinado en sucesivas iteraciones (dentro de los procesos de MÉTRICA de planificación y desarrollo, quedándose totalmente definidos tras el análisis y diseño)

- Actividad 5: Valoración del riesgo. El riesgo generalmente tiene que ser determinado específicamente para cada aplicación, y teniendo siempre en cuenta que la meta final es conseguir el 100% de aceptación del riesgo. En primer lugar hay que valorar si las amenazas son relevantes

según el nivel de seguridad especificado por los objetivos de seguridad. Después estimar los riesgos de seguridad basándose en las amenazas relevantes, su probabilidad de ocurrencia y su potencial impacto negativo. Todo esto queda reflejado en el 'Documento de Valoración del Riesgo', que será refinado en sucesivas iteraciones (en los procesos de MÉTRICA de planificación y desarrollo, quedándose totalmente detallado tras el análisis y diseño), realizándose inicialmente un esbozo del riesgo existente. Para realizar esta valoración se pueden usar distintas metodologías, como por ejemplo el estándar ISO/IEC 13335 (1-5), conocido como GMITS, proporciona una guía en el uso del proceso gestión de riesgos. O en España, se puede usar MAGERIT v.2 y en Reino Unido CRAMM (CCTA Risk Analysis and Management Method). De esta manera, este análisis de riesgos nos permitirá conocer cómo se ve afectada la tolerancia de la Organización a los riesgos relativos a cada amenaza. Además, las partes interesadas ("stakeholders") tomarán parte en esta actividad.

- Actividad 6: Elicitación de requisitos de seguridad. El RRS puede ser usado de nuevo en este paso. Para cada amenaza extraída del repositorio, se puede encontrar uno o más paquetes de requisitos de seguridad asociados. El ingeniero de requisitos debe seleccionar los requisitos de seguridad o los paquetes de requisitos de seguridad más adecuados que mitiguen las amenazas al nivel necesario según la valoración del riesgo asociado a la misma. Sin embargo, puede que también se encuentren requisitos de seguridad o paquetes de requisitos de seguridad adicionales por otros medios. Además, debe de especificarse el test de seguridad para cada paquete de requisitos de seguridad así como las contramedidas para cada requisito de seguridad, aunque éstos sean refinados a nivel de diseño, haciéndose un esbozo de los mismos en la fase de análisis. Ya que coincidimos con Firesmith [6] en que se debería tener cuidado en evitar la especificación innecesaria y prematura de mecanismos arquitectónicos de seguridad. De esta manera, al final de esta actividad y tras la fase de análisis, de acuerdo con la ISO/IEC 17799:2005, serán especificados los requisitos de seguridad funcionales, de aseguramiento y organizativos, junto con los requisitos de seguridad del entorno de desarrollo y de operación del SI. Además, en esta actividad se genera el 'Documento de Especificación de Requisitos de Seguridad', el cual se refinará en sucesivas iteraciones hasta el análisis y diseño donde ya estará suficientemente completado y detallado.

- Actividad 7: Categorización y priorización de requisitos. Cada requisito es categorizado y priorizado de forma cualitativa. De tal manera que los requisitos más importantes (en términos de impacto y probabilidad de ocurrencia) sean gestionados primero.

- Actividad 8: Inspección de requisitos. La inspección de requisitos se realiza para validar los artefactos generados, así como los elementos modificados del modelo y los nuevos elementos generados. De manera que se genera un 'Informe de Validación', cuyo fin es la revisión de la calidad del trabajo realizado por el equipo y de los entregables generados. Se verifica si los requisitos de seguridad son conformes al estándar IEEE 830-1998 y se llevará acabo dentro de las actividades de aseguramiento de la calidad del ciclo de vida, ayudándose de los requisitos de aseguramiento de los CC y el SSE-CMM, "System Security Engineering Capability Maturity Model" (ISO/IEC 21827). Y esta inspección es realizada por el equipo de inspección y de calidad, de manera que sirva como comprobación final de la calidad de los resultados obtenidos en la iteración. Después de esto se redactará la documentación relativa a los requisitos de seguridad, de manera que se genera el 'Documento con la Base Lógica de los Requisitos de Seguridad', donde se muestra como si se satisfacen todos los requisitos funcionales, organizativos y de aseguramiento, y todos los objetivos de seguridad se cumplen, el problema de seguridad está resuelto: todas las amenazas son contrarrestadas, las políticas de seguridad de la Organización se cumplen y todas las suposiciones son soportadas.

- Actividad 9: Mejora del repositorio. Los nuevos elementos del modelo (amenazas, requisitos, etc...) generados a lo largo del desarrollo de los pasos anteriores y susceptibles de ser usados en futuras aplicaciones y con la suficiente calidad según el Informe de Validación, son introducidos en el RRS.

Además, los elementos del modelo ya existentes en el repositorio pueden ser modificados para mejorar su calidad. De esta manera, todos estos nuevos y modificados elementos / artefactos en la iteración constituirán una línea base.

Por último, al mismo tiempo que integramos en estas actividades los requisitos funcionales de los CC (dentro de la actividad de "Elicitación de requisitos de seguridad"), proponemos definir en el plan de pruebas los criterios de evaluación de los requisitos de seguridad y su nivel de aseguramiento (EAL) con el fin de evaluar el SI. Además, junto con esta evaluación, plantemos la evaluación del proceso de ingeniería de seguridad haciendo uso del estándar ISO/IEC 21827 (SSE-CMM) tal y como se propone en [12]. Para ello, paralelamente, planteamos introducir los requisitos de aseguramiento de los CC dentro de las actividades de las interfaces de seguridad, de calidad y de gestión de proyectos de MÉTRICA. Aunque esta exposición queda fuera del alcance de esta comunicación.

3.4. MÉTRICA v.3 y SREP

SREP es compatible con MÉTRICA v.3 y su interfaz de seguridad con MAGERIT v.2, de forma que SREP complementa a MÉTRICA, proporcionándola soporte principalmente para el establecimiento de requisitos de seguridad a las siguientes actividades: en EVS 3 "Definición de requisitos del sistema", las tareas EVS 3.2 y 3.3 "Identificación y Catalogación de requisitos"; ASI 2, "Establecimiento de Requisitos"; en ASI 9 "Análisis de consistencia y especificación de requisitos", la tarea ASI 9.4 "Elaboración de la especificación de requisitos software"; en DSI 1 "Definición de la arquitectura del sistema" la tarea DSI 1.7 "Especificación de requisitos de operación y seguridad"; ASI-SEG 2 "Descripción de las funciones y mecanismos de seguridad", ASI-SEG 3 "Definición de los criterios de aceptación de la seguridad", DSI-SEG 2 "Especificación de requisitos de seguridad del entorno tecnológico", DSI-SEG 3 "Requisitos de seguridad del entorno de construcción". También proporciona apoyo planteando el uso de nuevas técnicas y artefactos para el tratamiento de requisitos de seguridad en las anteriores actividades: como un repositorio de recursos de seguridad, los casos de uso de seguridad, UMLSec y los casos de mal uso. Como se puede observar SREP complementa fundamentalmente a MÉTRICA v.3 en las primeras fases, especialmente en el ASI y el DSI, facilitando el establecimiento de requisitos de seguridad. Pero además SREP apoyará en el tratamiento de los requisitos de seguridad durante todo el ciclo de vida a algunas otras actividades a parte de las anteriores, como por ejemplo aquellas actividades relacionadas con las pruebas de seguridad, y también se podrán aprovechar de sus técnicas y artefactos, como por ejemplo del uso del repositorio de recursos de seguridad, aunque dicha exposición queda fuera del alcance de esta comunicación.

Por último, tal como se ha expuesto en el modelo de proceso de SREP, en lo relativo a MAGERIT v.2 se puede observar cómo ésta es complementaria a SREP, ya que puede usarse como metodología para realizar el análisis de riesgos que en SREP se plantea.

4. Conclusiones

Hoy en día, en la llamada Sociedad de la Información, dada la criticidad creciente de los SI unido a los nuevos requisitos legales y gubernamentales, se hace necesario el desarrollo de enfoques cada vez más sofisticados para asegurar la seguridad de la información. Normalmente, la seguridad de la información se abordaba desde el punto de vista técnico en la fase de implementación, y aunque éste se trata de un aspecto importante, consideramos fundamental el tratamiento de la seguridad en todas las fases del desarrollo de SI, especialmente en el establecimiento de los requisitos de seguridad, ya que constituyen la base para la consecución de un SI robusto.

Por ello presentamos una propuesta que viene a complementar a MÉTRICA v.3 y su interfaz de seguridad con MAGERIT v.2 en el tratamiento de los requisitos de seguridad en las primeras fases de desarrollo del software de una forma sistemática e intuitiva, para ello se basa en la reutilización de requisitos de seguridad, proporcionando un Repositorio de Recursos de Seguridad (RRS). Y se apoya también en la integración de los Criterios Comunes en el modelo de ciclo de vida de desarrollo del software. Además, dicha propuesta es conforme a los estándares ISO/IEC 15408 e ISO/IEC 17799:2005. El núcleo de SREP se fundamenta en el concepto de la construcción iterativa de software, proponiéndose un micro-proceso para el análisis de requisitos de seguridad, compuesto por nueve actividades, que son realizadas repetidamente en cada nivel de abstracción a lo largo del desarrollo incremental, pero con distinto énfasis según en la fase en la que se esté en el ciclo de vida. Finalmente, uno de los aspectos más relevantes de nuestra propuesta es que integra otros enfoques o técnicas, como SIREN [21], UMLSec [19], los casos de uso de seguridad [7] o los casos de mal uso [20].

Por ultimo, se ha realizado un refinamiento teórico del proceso, detallándose y completándose más, mediante su aplicación a casos prácticos, siendo alguno relacionado con la administración electrónica y siendo por tanto muy similar al que se presenta en MAGERIT v. 2.

5. Agradecimientos

Esta comunicación ha sido desarrollada en el contexto de los proyectos DIMENSIONS (PBC-05-012-2) Proyecto de la Consejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla- La Mancha y el FEDER, y los proyectos CALIPO (TIC2003-07804-C05-03) y RETISTIC (TIC2002-12487-E) de Dirección General de Investigación del Ministerio de Ciencia y Tecnología.

6. Referencias Bibliográficas

1. Baskerville, R., The development duality of information systems security. Journal of Management Systems, 1992. 4(1): p. 1-12.

2. Breu, R., Burger, K., Hafner, M., and Popp, G., Towards a Systematic Development of Secure Systems. 2004: WOSIS 2004.

3. CERT, http://www.cert.org.

4. Cybulsky, J. and Reed, K., Requirements Classification and Reuse: Crossing Domains Boundaries. ICSR'2000, 2000: p. 190-210.

5. Fernández-Medina, E., Moya, R., and Piattini Velthus, M., Gestión de Requisitos de Seguridad, in Seguridad de las Tecnologías de la Información "La construcción de la confianza para una sociedad conectada", AENOR, Editor. 2003. p. pp 593-618.

6. Firesmith, D.G., Engineering Security Requirements. Journal of Object Technology, 2003. 2(1): p. 53-68.

7. Firesmith, D.G., Security Use Cases. 2003: Journal of Object Technology. p. 53-64.

8. Kam, S.H., Integrating the Common Criteria Into the Software Engineering Lifecycle. IDEAS'05, 2005: p. 267-273.

9. Kemmerer, R., Cybersecurity. Proc. ICSE'03- 25th Intl. Conf. on Software engineering, 2003: p. 705-715.

10. Kim., H.-K., Automatic Translation Form Requirements Model into Use Cases Modeling on UML, C. Youn-Ky, Editor. 2005: ICCSA 2005.

11. Kotonya, G. and Sommerville, I., Requirements Engineering Process and Techniques. 1998.

12. Lee, J., Lee, J., Lee, S., and Choi, B., A CC-based Security Engineering Process Evaluation Model. 27th Annual International Computer Software and Applications Conference (COMPSAC'03), 2003.

13. McDermott, J. and Fox, C. Using Abuse Case Models for Security Requirements Analysis. in Annual Computer Security Applications Conference. 1999. Phoenix, Arizona.

14. Mead, N.R. and Stehney, T., Security Quality Requirements Engineering (SQUARE) Methodology. ICSE, 2005.

15. 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) - (accepted) -, 2006.

16. Mellado, D., Fernández-Medina, E., and Piattini, M., A Comparison of the Common Criteria with Proposals of Information Systems Security Requirements. FARES - International Symposium on Frontiers in Availability, Reliability and Security, in conjunction with The First International Conference on Availability, Reliability and Security (ARES 2006) - (accepted) -, 2006.

17. Mouratidis, H., Giorgini, P., Manson, G., and Philp, I. A Natural Extension of Tropos Methodology for Modelling Security. in Workshop on Agent-oriented methodologies, at OOPSLA 2002. 2003. Seattle, WA, USA.

18. Nuseibeh, Weaving Together Requirements and Architectures. IEEE Computer, 2001: p. 115-117.

19. 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.
20. Sindre, G., Firesmith, D.G., and Opdahl, A.L., A Reuse-Based Approach to Determining Security Requirements. REFSQ'03, 2003.

21. Toval, A., Nicolás, J., Moros, B., and García, F., Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach. 2001: Requirements Engineering Journal. p. 205-219.