Revista Cubana de Ciencias Informáticas
Vol. 12, No. Especial UCIENCIA, Septiembre 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 205-221
http://rcci.uci.cu

Tipo de artículo: Artículo original
Temática: Ingeniería y Calidad de Software
Recibido: 22/05/2018 | Aceptado: 10/09/2018

Requisitos de Seguridad para aplicaciones web

Security Requirements for web applications

Yisel Niño Benitez [0000-0001-7567-1501]*, Nemury Silega Martínez [0000-0002-8436-5650]

Universidad de las Ciencias Informáticas. Carretera a San Antonio de los Baños, Km. 2 ½. Torrens, La Lisa, La Habana, Cuba. {ynino, nsilega}@uci.cu

* **Autor para correspondencia**: ynino@uci.cu

Resumen

El ritmo vertiginoso de los procesos de desarrollo de software actuales incrementa el riesgo de presentar vulnerabilidades en un sistema de software. El aseguramiento de la información y de los sistemas que la procesan es, por tanto, un objetivo crucial para las organizaciones. La gestión de la Seguridad Informática desde el inicio del desarrollo de software evita que los mecanismos de seguridad deban ser ajustados dentro de un diseño ya existente, lo que provocaría cambios que generalmente generan vulnerabilidades en el software, y un incremento de costo y el tiempo para solucionarlos. Sin embargo, un dilema común que encuentran los ingenieros de software durante el desarrollo de un Sistema es la falta de requerimientos de seguridad que permitan darle seguimiento desde etapas tempranas. En el trabajo se exponen varios elementos sobre el marco teórico referente a la Seguridad Informática y la Ingeniería de Requisitos. Además, se describe una propuesta preliminar de Requisitos No Funcionales de Seguridad para el desarrollo de aplicaciones web en la Universidad de las Ciencias Informáticas con el objetivo de reducir las vulnerabilidades.

Palabras clave: ingeniería de requisitos, propuesta, requisito no funcional seguridad, seguridad informática.

Abstract

The vertiginous pace of current software development processes increases the risk of presenting vulnerabilities in a software system. The assurance of information and the systems that process it is, therefore, a first level objective for organizations. The management of Computer Security since the beginning of software development prevents security mechanisms from being adjusted within an existing design, which would cause changes that usually translate into software vulnerabilities, and an increase in budget costs and time to solve them once they have been identified. A

common dilemma faced by software engineers in building a system is the lack of security requirements to manage them since early states. In *the work several elements are exposed on the theoretical framework of Computer Security and Requirements Engineering, as well as a first proposal of Non-Functional Security Requirements for the development of web applications at the University of Informatics Sciences in order to achieve the decrease in their vulnerabilities.*

Keywords: *informatic security, non-functional safety requirement, proposal, requirements engineering.*

Introducción

"En pocos años la Web ha evolucionado enormemente: se ha pasado de páginas sencillas, con pocas imágenes y contenidos estáticos a páginas complejas con contenidos dinámicos que provienen de bases de datos, lo que posibilita la creación de aplicaciones web" (Mora, 2002). La Seguridad Informática (SI en lo adelante) es el área que se enfoca en "la protección de la infraestructura computacional y todo lo relacionado con esta, especialmente, la información contenida o circulante" (CDI, 2017). La SI llega a ser un área de vital importancia dentro de la Ingeniería de Software (IS), ya que como sucede con la seguridad aplicada a otros entornos, trata de minimizar los riesgos asociados al acceso y utilización de determinado sistema de forma no autorizada y en general malintencionada.

"La IS, por su parte, abarca un proceso, una colección de métodos (prácticas) y un conjunto de herramientas que permiten a los profesionales construir software de alta calidad" (Roger S. Pressman, 2015). Peter Neur y Brian Randell, por su parte planteaban que "un objetivo importante de la ingeniería de software es permitir a los desarrolladores construir sistemas que operen de manera confiable proporcionando teorías, métodos y herramientas para el desarrollo de software de calidad" (Naur & Randell, 1968). Dentro de la IS, el conjunto de procesos, tareas y técnicas que permiten la definición y gestión de los requisitos de un producto, de un modo sistemático, es conocido como Ingeniería de Requisitos (IR). La IR incluye las actividades relacionadas con la determinación de las necesidades o de las condiciones a satisfacer para hacer un software nuevo o modificado, siendo una colección estructurada de actividades, mediante las cuales se obtienen, validan y mantienen documentados los requisitos del usuario y del sistema (Huebe, 2005).

Los requisitos del sistema se clasifican en funcionales (RF) y no funcionales (RNF). Los funcionales describen lo que un sistema debe hacer, mientras que los no funcionales se refieren directamente a las propiedades emergentes del sistema como fiabilidad, rendimiento, mantenibilidad, seguridad, portabilidad, estándares a utilizar, entre otros (Sommerville, 2005). El RNF Seguridad se define como grado de protección de los datos, software y/o plataforma tecnológica de posibles pérdidas, actividades no permitidas o uso para propósitos no establecidos previamente

(Huamaní, 2015). A diferencia de otros RNF como la fiabilidad y el rendimiento, la seguridad no ha sido completamente integrada dentro del ciclo de vida de desarrollo y todavía es considerada después que el sistema ha sido diseñado (Rosado, Blanco, Sánchez, & Medina, 2009).

En encuestas aplicadas a diferentes roles involucrados en el proceso de desarrollo en la Universidad de las Ciencias Informáticas (UCI), estos conocen que se definen RNF de Seguridad y el 100% de los encuestados lo considera imprescindible para el desarrollo de las aplicaciones seguras. Sin embargo, sobre la gestión de este RNF: solamente el 30% considera la trazabilidad y gestión de la seguridad en todo el ciclo de vida del proceso de desarrollo del producto, el 55% lo propone a partir de la disciplina Requisitos, a pesar de que solo un 10% de estos le da seguimiento hasta la disciplina de Análisis y Diseño, el resto no considera necesario el seguimiento en las disciplinas siguientes. El 95% del total conoce los inconvenientes de no hacer un tratamiento certero de la seguridad en el desarrollo e identifican como aspectos negativos en este sentido: penetración en los sistemas, uso indebido de los servidores y su información, uso indebido de credenciales de autenticación, inyecciones SQL, atraso en el cronograma de entrega del producto por la resolución de No Conformidades (NC) de seguridad en las pruebas de liberación y aumento del costo de desarrollo, imposibilidad de despliegue del producto en determinado entorno por no tener en cuentas las restricciones legales de seguridad del cliente y con ello la pérdida del prestigio de la entidad desarrolladora ante el cliente.

La encuesta aplicada ofrece evidencias empíricas que demuestran las insuficiencias en la gestión del RNF de seguridad. Además, se constató el impacto negativo de gestión ineficiente del RNF para el desarrollo de los sistemas en términos de calidad y tiempo. Con el objetivo de aliviar la problemática descrita anteriormente en este artículo se presenta una propuesta de RNF seguridad. Una característica clave de esta propuesta es que plantea el seguimiento del RNF de seguridad desde etapas tempranas lo que contribuirá a disminuir el número de vulnerabilidades en las aplicaciones web.

Análisis conceptual de elementos de la Seguridad Informática

La SI tiene el objetivo de minimizar los riesgos asociados al acceso y utilización de determinado sistema de forma no autorizada y en general malintencionada (Garfinkel, 1999). Estrada, Alba y Martín la definen como las características y condiciones de sistemas de procesamiento de datos y su almacenamiento, para garantizar su confidencialidad, integridad y disponibilidad (Estrada, Alba, & Martín, 2012). Aguilera (Aguilera López, 2010) la define como “la disciplina que se ocupa de diseñar las normas, procedimientos, métodos y técnicas destinados a conseguir un sistema de información (o informático) seguro y confiable”. Para la autora de la investigación, la SI se enfoca en minimizar

los riesgos y vulnerabilidades en los recursos de hardware y software relacionados con el acceso y la utilización malintencionada de la información de los sistemas de software, para garantizar la integridad, confidencialidad y disponibilidad de la misma.

Para afrontar el establecimiento de un sistema de seguridad es necesario conocer (Aguilera López, 2010):

- **cuáles son los elementos que componen el sistema**: esta información se obtiene mediante entrevistas con los responsables o directivos de la organización, para lo que previamente hay que realizar un estudio de los riesgos que puedan presentar,
- **cuáles son los peligros que afectan al sistema, accidentalmente o provocados**: estos datos se deducen de los aportados tanto por la organización como por el estudio y prueba del propio sistema,
- **cuáles son las medidas que deberían adoptarse para** conocer, prevenir, impedir, reducir y controlar los riesgos potenciales, definiendo los servicios y mecanismos necesarios para minimizarlos.

La SI se podría resumir, en cinco principios fundamentales (CCM, 2016), los tres primeros también relacionados en la ISO 27002:2013 y en la NC ISO/IEC 25010:2016 (O. N. d. Normalización, 2016) y comunes en (Aguilera López, 2010), (Estrada et al., 2012), (Garfinkel, 1999):

- **Integridad**: garantiza que los datos no sean modificados desde su creación sin autorización y que ningún intruso pueda capturar y modificar los datos en tránsito.
- **Confidencialidad**: garantiza que la información, almacenada en el sistema informático o transmitida por la red, solamente va a estar disponible para aquellas personas autorizadas a accederla.
- **Disponibilidad**: garantiza el correcto funcionamiento de los sistemas de información y su disponibilidad en todo momento para los usuarios autorizados.
- **No repudio**: garantiza la participación de las partes en una comunicación. El uso y/o modificación de la información por parte de un usuario debe ser irrefutable, es decir, que el usuario no puede negar dicha acción.
- **Autenticación o Autenticidad**: asegura que sólo los individuos autorizados tengan acceso a los recursos.

Para normar estos principios atendiendo a los elementos que puedan componer un sistema, las regulaciones de cada entidad y los riesgos que puedan presentarse en su entorno, se han especificado estándares que tienen como objetivo minimizar estos riesgos y las vulnerabilidades que puedan aparecer en un escenario determinado en cuanto a SI.

Revista Cubana de Ciencias Informáticas
Vol. 12, No. Especial UCIENCIA, Septiembre 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 205-221
http://rcci.uci.cu

Estándares de Seguridad Informática

Ante la amenaza de ataques informáticos, las organizaciones deben demostrar que realizan una gestión competente y efectiva de la seguridad de los recursos y datos que gestionan. Este aspecto hace necesario el uso de estándares o normas que le orienten de forma estructurada, sistemática y coherente cómo proceder ante una situación de este tipo y fundamentalmente en su prevención, teniendo en cuenta que "las personas solo pueden hacer lo correcto si saben lo que es correcto" (OWASP, Meucci, & Muller, 2014)

- **ISO 17799** ofrece recomendaciones para realizar la gestión de la seguridad de la información dirigidas a los responsables de iniciar, implantar y mantener la seguridad de una organización. Define la información como un activo que posee valor y requiere por tanto de una protección adecuada (ISO, 2005).
- **La familia de normas ISO/IEC 27000** es un conjunto de estándares de seguridad que proporciona un marco para la gestión de la seguridad. Contiene buenas prácticas recomendadas en seguridad de la información para desarrollar, implementar y mantener especificaciones para los Sistemas de Gestión de la Seguridad de la Información (SGSI) (ISO/IEC, 2005b) y (Mentor, 2017). A continuación, se profundiza en algunas de las normas de esta familia por considerarse relevantes para la investigación. El resto de las normas, sirven de apoyo a la interpretación y evaluación de estas tres primeras (N. O. N. d. Normalización, 2007) y (Mentor, 2017):
- **ISO/IEC 27000: 2014 Tecnología de la información - Técnicas de seguridad - Sistemas de gestión de la seguridad de la información - Generalidades y vocabulario**: proporciona una visión general de las normas que componen la serie 27000, indicando para cada una de ellas su alcance y propósito. Recoge todas las definiciones para la serie de normas 27000 y aporta las bases de por qué es importante la implantación de un SGSI, una introducción a estos y una breve descripción de los pasos para su establecimiento, monitorización, mantenimiento y mejora.
- **ISO/IEC 27001:2013 Tecnología de la información - Técnicas de seguridad - Sistemas de gestión de seguridad de la información - Requisitos**: especifica los requisitos para establecer, implementar, mantener y mejorar continuamente un SGSI dentro del contexto de la organización. Los requisitos establecidos en ISO/IEC 27001: 2013 son genéricos y están destinados a ser aplicables a todas las organizaciones, independientemente de su tipo, tamaño o naturaleza. Esta norma tiene un enfoque a procesos.

- **ISO/IEC 27002:2013 Tecnología de la información - Técnicas de seguridad - Código de prácticas para los controles de seguridad de la información**: antigua ISO 17799:2005. Es una guía de buenas prácticas que describe los objetivos de control y controles recomendables en cuanto a seguridad de la información. Cuenta con 14 Dominios, 35 Objetivos de Control y 114 Controles. Se diferencia de la anterior (27001) que está enfocada a controles y no a procesos.
- "**OWASP (Open Web Application Security Project)** es una organización sin fines de lucro a nivel mundial 501 (c) (3) enfocada en mejorar la seguridad del software". Su misión es hacer visible la seguridad del software, para que las personas y las organizaciones sean capaces de tomar decisiones al respecto. OWASP proporciona información imparcial y práctica sobre aplicaciones seguras a individuos, corporaciones, universidades, agencias gubernamentales y otras organizaciones en todo el mundo (OWASP, 2017a). Emite herramientas de software y documentación basadas en el conocimiento sobre la seguridad de las aplicaciones (OWASP, 2017a). Como parte de estas herramientas publica cada cierto tiempo el Top 10 de riesgos más críticos y el Top 10 de controles proactivos a tener en cuenta en las aplicaciones de software. El objetivo de estos programas es crear conciencia sobre la seguridad de la aplicación al describir las áreas de preocupación más importantes en las que los desarrolladores de software deben estar conscientes (OWASP, 2016).

En el estudio realizado por Katy Anton, Jim Bird, Jim Manico (OWASP, 2016) se describen una lista de conceptos de seguridad que deben incluirse en cada proyecto de desarrollo de software. Los controles especificados en este documento se ordenan de acuerdo a su importancia, siendo el número 1 el más importante (OWASP, 2016) y (Brito, 2017). **La verificación de la seguridad temprano y frecuentemente** (Control 1 del Top 10 de Controles Proactivos) propone que desde el proceso de concepción del software se tengan en cuenta los requisitos de seguridad mientras se describen los requisitos del sistema, siempre teniendo en cuenta el resto de los controles propuestos. De igual manera propone verificar la seguridad con anticipación y a menudo en el proceso de desarrollo, ya sea a través de pruebas manuales o de pruebas y análisis automatizados. La gestión temprana de los requisitos de seguridad ayuda a evitar la ocurrencia de riesgos y vulnerabilidades en el proceso de desarrollo del software.

Estándares de Seguridad Informática en Cuba

Teniendo en cuenta el creciente avance de la informática en todas las esferas del desarrollo científico - técnico, económico, político y social, así como el surgimiento de nuevos riesgos asociados principalmente con el uso de las redes de datos de alcance global, en Cuba se han adoptado medidas de tipo legal que permiten regular la SI.

La Oficina de Seguridad para las Redes Informáticas (OSRI), tendrá como objeto social "llevar a cabo la prevención, evaluación, aviso, investigación y respuesta a las acciones, tanto internas como externas, que afecten el normal funcionamiento de las tecnologías de la información del país" ((OSRI), 2018). También se cuenta en Cuba con el centro de Ciberseguridad del Espacio, que es una "estructura especializada que contribuye al fortalecimiento de la seguridad en el ciberespacio cubano, fomentando la cooperación entre todos los factores que inciden en la ciberseguridad a nivel nacional y potenciando la colaboración internacional en esta esfera. Tiene como misión contribuir al fortalecimiento de la seguridad en el ciberespacio cubano y coordinar de manera efectiva la gestión de los eventos cibernéticos que impactan en la ciberseguridad de la nación" (CSC, 2018).

A partir de las vulnerabilidades y debilidades propias de los sistemas informáticos, de las dificultades y limitaciones que se presentan para detectar y neutralizar oportunamente las posibles acciones enemigas en esta esfera, se implementó un basamento legal que establece los requisitos de seguridad en el empleo de las tecnologías de la información a partir de criterios de racionalidad y utilidad, que resulten susceptibles de verificación y tienden a la disminución de los riesgos en la SI (MINCOM, 2007). Para ello el MINCOM estableció la Resolución 127/2007 que tiene por objetivo "establecer los requisitos que rigen la seguridad de las tecnologías de la información y garantizar un respaldo legal que responda a las condiciones y necesidades del proceso de informatización del país" (MINCOM, 2007). Además, se cuenta con la traducción normalizada por la Oficina Nacional de Normalización de la NC-ISO/IEC 27001: 2007, elaborada por el Comité Técnico de Normalización NC/CTN 18 de Tecnología de la Información.

Sin embargo, al considerar la SI solo en ciertas etapas del proceso de desarrollo, podrían provocarse conflictos entre las necesidades de seguridad y los RF del sistema. Tener en cuenta la seguridad junto con los RF del sistema a través de las etapas de desarrollo, ayudaría a limitar los casos de conflicto, identificándolos pronto en el desarrollo del sistema, y encontrando formas de superarlos (Rosado et al., 2009).

Ingeniería de requisitos

Pressman plantea que un proceso sólido de IR ayuda a garantizar que se ha especificado un sistema que recoge las necesidades del cliente y cumple con sus expectativas. Identifica como parte de este proceso siete tareas fundamentales: inicio (identificación de la necesidad u oportunidad de negocio), elicitación, elaboración, negociación, especificación, validación y administración de requisitos (Roger S. Pressman, 2015). Para Ian Sommerville en (Sommerville, 2005), la IR es el proceso de desarrollar una especificación del software. Las especificaciones pretenden comunicar las necesidades del sistema del cliente a los desarrolladores. Describe la IR en 4 pasos, estos tratan de la evaluación de si el sistema es útil para el negocio (estudio de viabilidad); el descubrimiento de requisitos

(obtención y análisis); la transformación de estos requisitos en formularios estándar (especificación), y la verificación de que los requisitos realmente definen el sistema que quiere el cliente (validación).

El término **requisito** puede conceptualizarse según lo planteado en la traducción certificada de la ISO 9000: 2000 como "la necesidad o expectativa establecida, generalmente implícita u obligatoria". Pueden utilizarse calificativos para identificar un tipo específico de requisito, por ejemplo, requisito de un producto, requisito de la gestión de la calidad, requisito del cliente. Los requisitos pueden ser generados por las diferentes partes interesadas (ISO, 2000). La IEEE publicó que un requisito es: "una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo. Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal" (IEEE, 1990). El SEI (del inglés Software Engineering Institute) plantea como parte del glosario del modelo CMMI que un requisito es: (1) Una condición o capacidad necesitada por un usuario para solucionar un problema o lograr un objetivo. (2) Una condición o capacidad que debe cumplir o poseer un producto o componente de producto para satisfacer un contrato, un estándar, una especificación u otros documentos impuestos formalmente. (3) Una representación documentada de una condición o capacidad como en (1) o en (2) (SEI, 2010).

Clasificación de requisitos

Sommerville clasifica los requisitos en dos categorías (Sommerville, 2005):

- **Requisitos del usuario**: son declaraciones, en lenguaje natural y en diagramas, de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar.
- **Requisitos del sistema**: establecen con detalle las funciones, servicios y restricciones operativas del sistema. El documento de requisitos del sistema (algunas veces denominado especificación funcional) debe ser preciso. Debe definir exactamente qué es lo que se va a implementar.

Para la Software Engineering Body of Knowledge (SWEBOK) existen dos grandes categorías en las que pueden clasificarse los requisitos (ISO/IEC, 2005a), estas son:

- **RF**: especifican acciones que el sistema debe ser capaz de realizar, sin tomar en consideración ningún tipo de restricción física. Especifican el comportamiento de entrada y salida del sistema y surgen de la razón fundamental de la existencia del producto. Indican características y restricciones sobre la funcionalidad del software. Definen el comportamiento interno del sistema.

- **RNF**: son propiedades o cualidades que el producto debe tener, también son conocidos como atributos de calidad. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable; normalmente están vinculados a RF.

Los requisitos de seguridad, se identifican mediante una evaluación metódica de los riesgos de seguridad (ISO/IEC, 2005b). Existen tres fuentes principales de requisitos de seguridad, una fuente se deriva de **evaluar los riesgos para la organización**, tomando en cuenta la estrategia general y los objetivos de esta. A través de la evaluación del riesgo, se identifican las amenazas para los activos, se evalúa la vulnerabilidad y la probabilidad de ocurrencia y se calcula el impacto potencial. Otra fuente son los **requisitos legales, reguladores, estatutarios y contractuales** que tienen que satisfacer una organización, sus socios comerciales, contratistas y proveedores de servicio; y su ambiente socio-cultural. Y una tercera es el **conjunto particular de principios, objetivos y requisitos comerciales** para el procesamiento de la información que una organización ha desarrollado para sostener sus operaciones (ISO/IEC, 2005b) y (Mentor, 2017).

Estándares y normas para la Ingeniería de Requisitos

Los estándares de calidad desarrollan normas y modelos en función de organizar y formalizar los procesos en la industria del software incluyendo los relativos a la IR.

- **ISO 29148 – Ingeniería de sistemas y software - Procesos del ciclo de vida - Ingeniería de requisitos** (ISO/IEC/IEEE, 2011): define la construcción de un buen requisito, proporciona atributos y características de este, y analiza la aplicación iterativa y recursiva de los procesos de requisitos a lo largo del ciclo de vida. Además, define los elementos de información aplicables a la IR y su contenido.
- **IEEE 830-1998 Prácticas recomendadas para la especificación de requisitos de software** (IEEE, 1998b): describe el contenido y las cualidades de una buena especificación, y también puede ser aplicada para ayudar en la selección de productos de software internos y comerciales. Define como características de una correcta especificación de requisitos que sean: correcto, sin ambigüedad, completo, consistente, clasificado por importancia y/o estabilidad, verificable, modificable, y trazable.
- **IEEE 1233-1998 Guía para desarrollar especificaciones de requisitos de sistema** (IEEE, 1998a): proporciona una guía para el desarrollo de las especificaciones de requisitos, que satisfará una necesidad expresada. Esta incluye la identificación, organización, presentación y modificación de los requisitos.

- **NC ISO/IEC 25010:2016 Ingeniería de software y sistemas – Requisitos de la calidad y evaluación de software (square) – Modelos de la calidad de software y sistemas** (N. ISO/IEC, 2016): define seis características de la calidad y describe un modelo de proceso de evaluación del producto de software. La seguridad se ha añadido como una característica, con las sub-características de confidencialidad, integridad, no repudio, responsabilidad y autenticidad.

Las normas ISO e IEEE son considerados estándares con un alto grado de complejidad lo que dificulta su entendimiento debido en su mayor parte a que se encuentran desagregadas en varios documentos que tienden a confundir a los interesados en su implementación y esto implica un aumento en los esfuerzos y costos para preparar la documentación e implantación de los sistemas. Estas normas no definen elementos fundamentales como una propuesta de proceso, qué roles deben ejecutar las actividades, cuáles técnicas o herramientas pueden utilizarse para apoyar el proceso. Aunque algunas hacen referencia a las mediciones, no realizan una propuesta detallada de indicadores concretos a medir y no se hace alusión a la gestión explícita del conocimiento.

"Los modelos CMMI® (Capability Maturity Model® Integration) son colecciones de buenas prácticas que ayudan a las organizaciones a mejorar sus procesos. Estos modelos son desarrollados por equipos con miembros procedentes de la industria, del gobierno y del Software Engineering Institute (SEI) (…) Las buenas prácticas del modelo se centran en las actividades para desarrollar productos y servicios de calidad con el fin de cumplir las necesidades de clientes y usuarios finales" (SEI, 2010).

CMMI define 22 áreas de procesos, concentradas en cuatro grandes grupos: gestión de procesos, gestión de proyectos, ingeniería y soporte. Las áreas de proceso de ingeniería cubren las actividades de desarrollo y de mantenimiento que se utilizan en todas las disciplinas de ingeniería y se aplican al desarrollo de cualquier producto o servicio dentro del dominio de desarrollo:

- Integración del Producto (PI).
- Desarrollo de Requisitos (RD).
- Solución Técnica (TS).
- Validación (VAL).
- Verificación (VER).

Revista Cubana de Ciencias Informáticas
Vol. 12, No. Especial UCIENCIA, Septiembre 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 205-221
http://rcci.uci.cu

Agregado a estas, dentro de las áreas de administración de proyecto se encuentra el área de Gestión de requisitos REQM, importante también en la IR y estrechamente relacionadas con las áreas del grupo ingenieril. De estas áreas de procesos se analizan las de REQM y RD específicamente por el impacto que tienen en la IR para la investigación:

- **REQM** (SEI, 2010): tiene como propósito gestionar los requisitos de los productos y los componentes de producto del proyecto, y asegurar la alineación entre esos requisitos, y los planes y los productos de trabajo del proyecto. El proyecto con la aplicación de REQM gestiona los cambios a los requisitos y su análisis razonado a medida que evolucionan e identifica inconsistencias que ocurren entre los planes, los productos de trabajo y los requisitos. Mantiene una trazabilidad bidireccional entre los requisitos con el resto de los productos de trabajo.
- **RD** (SEI, 2010): tiene como propósito "educir, analizar y establecer los requisitos de cliente, de producto y de componente de producto". Esta área de proceso tiene tres metas a cumplir: desarrollar los requisitos de cliente, desarrollar los requisitos de producto y analizar y validar los requisitos.

A partir de la importancia que se le confiere a la identificación temprana de los requisitos de seguridad y su seguimiento durante el ciclo de vida del proyecto, y teniendo en cuenta que la actividad productiva de la UCI está certificada con el nivel 2 de madurez de CMMI y se encuentra en el proceso de definición de los materiales para la certificación del nivel 3 se decide utilizar el modelo CMMI. Especialmente, las definiciones realizadas por la universidad de las áreas de procesos REQM y RD para la elaboración de una guía para la gestión del requisito no funcional seguridad, tomando como apoyo las definiciones de la ISO 25010 para el atributo de calidad seguridad. Para el análisis de la trazabilidad bidireccional de los requisitos se utilizará el subproceso definido por la universidad para el nivel 2 de CMMI. La trazabilidad ayuda a determinar si todos los requisitos fuente se han tratado totalmente y si todos los requisitos de nivel más bajo se pueden tracear hacia una fuente válida. Se hace además necesaria a la hora de evaluar el impacto de los cambios de los requisitos sobre las actividades del proyecto y los productos de trabajo resultantes. Como resultado de este proceso se obtienen matrices de trazabilidad entre los requisitos y los artefactos derivados en el proceso de desarrollo que aseguran el alineamiento entre el trabajo en el proyecto y los requisitos.

Requisitos de Seguridad para aplicaciones web

El alcance específico de la seguridad debe estar claramente definido por los interesados en términos de los activos a los que se aplica la seguridad y las consecuencias contra las que se evalúa la seguridad (NIST, ROSS, McEVILLEY, & OREN, 2016).

Teniendo en cuenta los resultados de encuestas aplicadas a diversos roles inmersos en el desarrollo de software pertenecientes a varias áreas de la UCI, a lo planteado por la Norma Ramal (NR) 2-1 Requisitos de la Calidad para Sistemas Informáticos y Productos de Software (CALISOFT, 2018), los Diez riesgos más críticos en Aplicaciones Web de OWASP (OWASP, 2017b), y el Estándar de Verificación de Seguridad en Aplicaciones de OWASP (OWASP, Manico, Stock, & Cuthbert, 2017) se identificaron un conjunto de requisitos de seguridad que deben gestionarse en el desarrollo de aplicaciones web en la UCI.

Los requisitos identificados serán agrupados de acuerdo a los principales objetivos de seguridad o sub-características analizados en la investigación:

Integridad:

- RNFS 1: utilizar marcos de trabajo que previenen automáticamente los ataques XSS (Cross-Site Scripting o inyección de código malicioso).
- RNFS 2: validar los datos que se reciben y velar por la integridad de los datos que se devuelven.
- RNFS 3: prevenir los ataques CSRF (del inglés Cross-Site Request Forgery o falsificación de petición en sitios cruzados).
- RNFS 4: evitar las inyecciones de código.
- RNFS 5: utilizar LIMIT y otros controles SQL para evitar la fuga masiva de datos en caso de inyecciones SQL.
- RNFS 6: validar la entrada de datos al servidor utilizando “listas blancas”.
- RNFS 7: cifrar los datos sensibles que sean almacenados.

Confidencialidad:

- RNFS 8: proteger las conexiones autenticadas o que involucren funciones o información relevante.
- RNFS 9: evitar mostrar referencias hacia objetos internos de la aplicación.
- RNFS 10: evitar mostrar mensajes con información que ayude a recopilar información sobre el producto o las configuraciones del servidor.
- RNFS 11: evitar la elevación de privilegios en las cuentas de usuarios.
- RNFS 12: revisar todos los elementos de la infraestructura para asegurar que no contengan ninguna vulnerabilidad conocida, así como las herramientas administrativas usadas para el mantenimiento de los diferentes componentes.
- RNFS 13: evitar almacenar datos sensibles de manera innecesaria.
- RNFS 14: deshabilitar el almacenamiento en caché de datos sensibles.

Disponibilidad:

- RNFS 15: realizar estudio sobre las posibles vulnerabilidades que se puedan presentar en la tecnología a utilizar en el desarrollo.
- RNFS 16: utilizar tecnologías seguras para el desarrollo.
- RNFS 17: cumplir los requisitos exclusivos de los límites de negocio de las aplicaciones.
- RNFS 18: controlar el receptor de escucha de las Bases de Datos.
- RNFS 19: garantizar que el servidor no envíe directrices o cabeceras de seguridad a los clientes o que se encuentren configurados con valores inseguros.
- RNFS 20: actualizar las configuraciones apropiadas de la tecnología usada de acuerdo a las advertencias de seguridad y seguir un proceso de gestión de parches.
- RNFS 21: utilizar una herramienta para mantener un inventario y control de versiones de los componentes
- RNFS 22: utilizar componentes únicamente de orígenes oficiales y utilizando los canales seguros.
- RNFS 23: analizar riesgos y vulnerabilidades del entorno de despliegue del cliente atendiendo a sus características.

No repudio:

- RNFS 24: cifrar todos los datos en tránsito utilizando protocolos seguros.
- RNFS 25: identificar o firmar de forma única los mensajes intercambiados.
- RNFS 26: almacenar los mensajes intercambiados en ficheros logs para su posterior consulta.

Autenticación o Autenticidad:

- RNFS 27: evitar mantener credenciales creadas por defecto, débiles o muy conocidas especialmente en el caso de los administradores del sistema.
- RNFS 28: definir mecanismos de autenticación personalizado para todos los usuarios del sistema.
- RNFS 29: evitar utilizar cuentas suministradas por defecto.
- RNFS 30: evitar ataques de fuerza bruta y/o ataques automatizados.
- RNFS 31: utilizar controles contra contraseñas débiles.
- RNFS 32: alinear la política de longitud, complejidad y rotación de las contraseñas establecidas.
- RNFS 33: limitar el tiempo de respuesta de cada intento fallido de inicio de sesión.
- RNFS 34: controlar el ciclo de vida de las contraseñas.

- RNFS 35: restringir el acceso de un usuario estándar (no administrador) a modificar sus privilegios en la aplicación o los de otro usuario con su mismo rol.
- RNFS 36: cerrar automáticamente la sesión de un usuario cuando ha estado inactivo durante un cierto lapso de tiempo.
- RNFS 37: destruir el ID de sesión luego de salir o cerrar el sistema.

A continuación, se muestran las relaciones entre varios de los requisitos con riesgos identificados por un equipo de investigadores de OWASP (OWASP, 2017b).

Tabla 1. Relación entre riesgos y vulnerabilidades con requisitos de seguridad

Riesgos/Vulnerabilidades	**Requisitos que se relacionan**
Inyección	RNFS 1, RNFS 2, RNFS 3, RNFS 4, RNFS 5, RNFS 6
Pérdida de autenticación	RNFS 27, RNFS 28, RNFS 29, RNFS 30, RNFS 31, RNFS 32, RNFS 33, RNFS 34, RNFS 35, RNFS 36, RNFS 37
Exposición de datos sensibles	RNF 7, RNFS 13, RNFS 14
Entidades externas XML	RNFS 6
Pérdida de control de acceso	RNFS 11, RNFS 17, RNFS 35
Configuración de seguridad incorrecta	RNFS 15, RNFS 16, RNFS 19, RNFS 20, RNFS 22
Ataques de XSS (Cross-Site Scripting)	RNFS 1, RNFS 2,
Uso de componentes con vulnerabilidades conocidas	RNFS 19, RNFS 20, RNFS 21, RNFS 22, RNFS 23

Aunque no se relacionan en la tabla anterior, el resto de los requisitos mitigan la aparición de vulnerabilidades que pueden estar presentes en el desarrollo de las aplicaciones web y atentar contra la seguridad del producto final. Con la identificación preliminar de los requisitos expuestos en la investigación, se pretende que el uso de estos contribuya a elevar el conocimiento en materia de SI y la calidad de las aplicaciones web desarrolladas en la UCI, lo que se garantiza desde etapas tempranas del desarrollo del producto, gracias a la tipificación de riesgos y/o vulnerabilidades que pueden estar presentes tanto en el entorno del cliente como en el equipo de desarrollo. Se tienen en cuenta, aspectos legales que igualmente deben garantizarse y que se pueden ver reflejados en los requisitos presentados en la investigación. Los requisitos propuestos deben formar parte de los resultados obtenidos desde la concepción de los procesos de negocio y deben ser traceados y monitoreados a medida que el desarrollo del sistema evolucione, lo que traería como resultado positivo una disminución de las posibles vulnerabilidades que pudiera tener el producto final.

Conclusiones

En el presente trabajo se hace una revisión de conceptos relevantes sobre SI. Este análisis permitió arribar a las siguientes conclusiones: la SI se enfoca en minimizar los riesgos existentes en el acceso y utilización mal intencionada de la información de los sistemas de software. La SI es un tema que recibe la atención de la industria de software lo que se puede evidenciar con la presencia de varios documentos tales como la familia de normas ISO/IEC 27000, diferentes materiales estandarizados de OWASP y la Resolución 127:2007 del MINCOM que ofrecen definiciones precisas para la identificación temprana de los requisitos de seguridad y el establecimiento de los SGSI. Estos documentos corroboran la importancia conferida a la identificación temprana de los requisitos de seguridad y su seguimiento durante todo el desarrollo del ciclo de vida del proyecto, tal y como propone OWASP en el Top 10 de controles proactivos 2016. La IR desempeña un rol primordial en los proyectos de desarrollo de software debido a que facilita los métodos y técnicas apropiadas para comprender lo que desea el cliente. Analiza, comprende y valida dichas necesidades, traducidas ya formalmente en requisitos de software. Se decidió utilizar las definiciones de CMMI en la UCI sobre las áreas de procesos de REQM y RD para la posterior elaboración de una guía para la gestión del RNF seguridad. Se realizó una propuesta preliminar de requisitos de seguridad basados en la Norma Ramal (NR) 2-1 Requisitos de la Calidad para Sistemas Informáticos y Productos de Software y los Diez riesgos más críticos en Aplicaciones Web de OWASP que a partir de su temprana y adecuada gestión permitirá la disminución de las posibles vulnerabilidades que pudiera tener el producto final.

Referencias

(OSRI, 2018), O. d. S. d. R. I. (2018). Retrieved 29/03/2018, 2018, from http://www.mincom.gob.cu/?q=node/311

Aguilera López, P. (2010). *Seguridad informática*. México.

Brito, H. R. G. (Producer). (2017, 03 20). Behique Digital. *Behique Digital*. Retrieved from https://henryraul.wordpress.com/2016/10/11/owasp-top-10-proactive-controls-2016/

CALISOFT, C. N. d. C. d. S. (2018). Norma Ramal – Requisitos de la Calidad para Sistemas Informáticos y Productos de Software. Retrieved 23/05/2018, 2018, from http://subcomite7.cubava.cu/2017/02/10/norma-ramal-requisitos-de-la-calidad-para-sistemas-informaticos-y-productos-de-software/

CCM. (2016). Introducción a la seguridad informática. Retrieved 31 mayo 2017, from http://es.ccm.net/contents/622-introduccion-a-la-seguridad-informatica

CDI (Producer). (2017, Enero 24). Centro de Delitos Informáticos. *Centro de Delitos Informáticos*. Retrieved from https://centrodelitosinformaticos.com/seguridad-informatica/

CSC, C. d. C. d. E. (2018). Objetivos y Misión. from http://www.cscuba.cu/es/node/221

Estrada, Y., Alba, W., & Martín, A. (2012). Fundamentos para implementar y certificar un Sistema de Gestión de la Seguridad Informática bajo la Norma ISO/IEC 27001. *Serie Científica de la Universidad de las Ciencias Informáticas, No. 10, Vol. 5*, 10.

Garfinkel, S. (1999). *Seguridad y Comercio en la Web*: McGraw Hill/Interamericana de España.

Huamaní, I. Y. M. (2015). *SISTEMA DE INFORMACIÓN PARA EL PROCEDIMIENTO DE GESTIÓN DE SERVICIOS ARBITRALES EN CONTRATACIONES DEL ESTADO.* UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR, Villa El Salvador.

Huebe, M. d. L. P. (2005). *Ingeniería de sistemas.* Universidad Autónoma del Estado de Hidalgo. , Pachuca.

IEEE. (1990). *IEEE Terminología estándar de ingeniería de software*.

IEEE. (1998a). *IEEE Guía para desarrollar especificaciones de requisitos del sistema*.

IEEE. (1998b). *IEEE Std 830-1998 Práctica recomendada para especificaciones de requisitos de software*.

ISO. (2000). *ISO 9000:2000 Sistemas de gestión de la calidad — Conceptos y vocabulario*.

ISO. (2005). *ISO/IEC 17799 Tecnologías de la información - Técnicas de seguridad - Código para la práctica de la gestión de la seguridad de la información*

ISO/IEC. (2005a). Ingeniería de Software - Guía del Cuerpo de Conocimiento de Ingeniería de Software (SWEBOK).

ISO/IEC. (2005b). ISO/IEC 27001 Tecnología de la información - Técnicas de seguridad - Sistemas de gestión de seguridad de la información - Requisitos: Switzerland.

ISO/IEC, N. (2016). INGENIERÍA DE SOFTWARE Y SISTEMAS – REQUISITOS DE LA CALIDAD Y EVALUACIÓN DE SOFTWARE (SQuaRE) – MODELOS DE LA CALIDAD DE SOFTWARE Y SISTEMAS

(ISO/IEC 25010: 2011, IDT).

ISO/IEC/IEEE. (2011). *Ingeniería de software y de sistemas - Procesos del ciclo de vida - Ingeniería de requisitos*.

Mentor, A. (Producer). (2017, marzo 12). Aula Mentor. *Aula Mentor. Seguridad informática. Normas ISO sobre gestión de seguridad de la información*. Retrieved from http://descargas.pntic.mec.es/mentor/visitas/demoSeguridadInformatica/normas_iso_sobre_gestin_de_seguridad_de_la_informacin.html

RESOLUCION No. 127 /2007 (2007).

Mora, S. L. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web* (E. C. Universitario Ed.).

Naur, P., & Randell, B. (1968). Ingeniería de software (pp. 136). Report on a conference sponsored by the NATO SCIENCE COMMITTEE Garmisch, Germany.

NIST, ROSS, R., McEVILLEY, M., & OREN, J. C. (2016). INGENIERÍA DE SEGURIDAD DE SISTEMAS - Consideraciones para un Enfoque Multidisciplinario en la Ingeniería de Sistemas Confiables Confiables.

Normalización, N. O. N. d. (2007). ISO/IEC 27001: 2007 TECNOLOGÍA DE LA INFORMACIÓN—TÉCNICAS DE SEGURIDAD—SISTEMAS DE GESTIÓN DE LA SEGURIDAD DE LA INFORMACIÓN—REQUISITOS (ISO/IEC 27001: 2005, IDT): Cuban National Bureau of Standards.

Normalización, O. N. d. (2016). *INGENIERÍA DE SOFTWARE Y SISTEMAS – REQUISITOS DE LA CALIDAD Y EVALUACIÓN DE SOFTWARE (SQuaRE) – MODELOS DE LA CALIDAD DE SOFTWARE Y SISTEMAS (ISO/IEC 25010: 2011, IDT)*. La Habana, Cuba.

OWASP. (2016). OWASP Top 10 controles proactivos 2016. 28.

OWASP (Producer). (2017a, 03 21). OWASP. *OWASP*. Retrieved from https://www.owasp.org/index.php/Main_Page

OWASP. (2017b). OWASP Top 10 - 2017 Los diez riesgos más críticos en Aplicaciones Web.

OWASP, Manico, J., Stock, A. v. d., & Cuthbert, D. (2017). Estándar de Verificación de Seguridad en Aplicaciones 3.0.1.

OWASP, Meucci, M., & Muller, A. (2014). *OWASP Guía para probadores 4.0*

Roger S. Pressman, B. R. M. (2015). *Ingeniería de software. Un enfoque práctico. 8ª edición*. New York.

Rosado, D. G., Blanco, C., Sánchez, L. E., & Medina, E. F. (2009). La Seguridad como una asignatura indispensable para un Ingeniero del Software. La Mancha.

SEI. (2010). CMMI® para Desarrollo, Version 1.3: Carnegie Mellon University.

Sommerville, I. (2005). *Ingeniería del software. 7ma Edición*. United Kingdom: Pearson Education.