DESARROLLO DE PRODUCTOS DE SOFTWARE SEGUROS EN SINTONÍA CON LOS MODELOS SSE-CMM, COBIT E ITIL

Edmundo Tovar [2], José Carrillo [2], Vianca Vega [1], Gloria Gasca [2]
[1] *Universidad Católica del Norte*
E-Mail : vvega@ucn.cl
[2] *Universidad Politécnica de Madrid*
E-Mail : {etovar, jcarrillo }@fi.upm.es, glogasca@yahoo.com

Abstract: The components that form the Information Technology (IT) that have the organizations, every day take greater relevance, thus models as COBIT and ITIL have been developed, who guide the practices for their administration and control. Also, due to the relevance acquired by the security aspect and IT has been developed SSE-CMM that is a model for the security administration. On the other hand, it is of interest a particular element of Technology, the Software Applications, that must be developed considering from the beginnings of it life cycle, the security as an emergent characteristic. The present article, analyzes how these models contribute and harness the secure software development. In addition, a new definition of the dimensions appears to consider to reach the security of integral way, as an emergent property of the software systems and the coincident aspects between the models are emphasized, in relation to the software security.

Resumen: Los componentes que forman la Tecnología de Información (TI) que poseen las organizaciones, cada día toman mayor relevancia, por lo cual se han desarrollado Modelos que guían las prácticas para su administración y control, tal es el caso de COBIT e ITIL. También, debido a la relevancia adquirida por el aspecto de seguridad y las TI's se ha desarrollado SSE-CMM que es un modelo para la administración de la seguridad. Por otro lado, es de interés un tipo de elemento particular de la Tecnología, las aplicaciones Software, que deben ser desarrolladas teniendo en cuenta desde los inicios de su ciclo de vida, la seguridad como una característica emergente. El presente artículo, analiza cómo estos modelos aportan y potencian el desarrollo de productos de software seguro. Además se presenta una nueva definición de las dimensiones a considerar para alcanzar la seguridad de manera integral, como una propiedad emergente de los sistemas de software y se resaltan los aspectos coincidentes entre los modelos, en relación a la seguridad del software.

Palabras Clave: Software Seguro, Modelos de Calidad, SSE-CMM, COBIT, ITIL

1. SEGURIDAD DEL SOFTWARE

La seguridad del software es una idea de la Ingeniería de Software que busca que un producto desarrollado continúe funcionando correctamente ante ataques maliciosos [1], y puede ser vista como una medida de robustez de un sistema de software, respecto a una política de seguridad [2]. Es decir, que la raíz de muchos problemas de seguridad está en que el software falla de forma inesperada [2].

Históricamente, la prensa popular presta mucha atención a los virus y a la denegación de servicios. Estos tópicos son importantes para la seguridad. Sin embargo la gran mayoría de los interesados no llegan al fondo del tema cuando informan sobre él. Detrás de cada problema de seguridad en un ordenador y de cada ataque malicioso hay un enemigo común: la baja calidad del software.

También se debe mencionar, que en ocasiones, el concepto de seguridad del software se confunde con aspectos de seguridad de la información, lo cual es incorrecto, dado que si bien es cierto, ambos enfoques se encuentran relacionados, se diferencian porque el enfoque que aquí se plantea contempla la seguridad del producto en todos sus aspectos, trascendiendo hasta el proceso de desarrollo, mientras que el objetivo de la seguridad de la información es proteger los productos software ya

desarrollados, vistos como un activo más de la TI, sin buscar soluciones a los problemas de seguridad que actualmente afectan al software.

Existen varios modelos y estándares desarrollados sobre la base de las características de calidad, bajo los cuales se consideran, por ejemplo, la mantenibilidad del software, reusabilidad, corrección, integridad, eficiencia y seguridad, entre otras y varían según el modelo.

En los últimos años, el interés en la seguridad del software se ha ido incrementando, surgiendo muchas investigaciones sobre el tema, pero aún falta lograr la aplicación e implementación de estas investigaciones [3], lo cual deja abiertos ámbitos de trabajo en torno al tema.

McGraw en [1] indica que la base de los problemas de seguridad son la conectividad, la complejidad y la extensibilidad de los sistemas actuales. La afirmación anterior se refiere a que actualmente la mayoría de los productos de software están disponibles en un medio tan abierto como Internet (conectividad), y además estos mismos productos son parte de sistemas complejos que apoyan objetivos de negocios que no son simples ni aislados (complejidad) y que al mismo tiempo estos objetivos de negocios deben irse adecuando a los constantes cambios que exigen los mercados actuales (extensibilidad). Como resultado de lo anterior, los problemas de seguridad del software se han ido incrementando, llegando en ocasiones, incluso a poner en riesgo el éxito y sobrevivencia de las organizaciones.

Los aspectos de seguridad en la Ingeniería de Software, van más allá de la construcción del producto y trascienden dentro del proceso de software, de ahí que la preocupación de diferentes organizaciones expresen el interés por crear y proponer estándares, metodologías y métodos que contemplan la seguridad desde el punto de vista del proceso y la enmarquen como una característica de calidad del software importante para su desarrollo, es el caso del planteamiento de la norma ISO 9126, un estándar internacional para la evaluación de la calidad de productos de software, con el nombre "Information Technology – Software Product Evaluation – Quality Characteristics and Guidelines for their use"[4]. Bajo esta misma perspectiva, y teniendo en consideración que los productos software son parte del inventario de tecnología de la información que las organizaciones poseen, el presente trabajo analiza los modelos SSE-CMM, COBIT E ITIL para determinar cómo se enlazan y reafirman las propuestas existentes para el desarrollo de productos de software seguro.

2. PRESENTACIÓN Y ANÁLISIS DE LAS METODOLOGÍAS

La seguridad es un problema muy serio y la tendencia actual consiste en descargar el problema de seguridad en la configuración de mecanismos externos al software que supuestamente lo protegen de los ataques; o justificar la seguridad del software bajo la seguridad de la información, confundiéndose dos aspectos diferentes. Un típico ejemplo de lo anterior es el uso de "firewall" y técnicas criptográficas, que si bien es cierto, pueden ser útiles a la hora de reforzar la seguridad de un producto software, no son suficientes por si mismos si la aplicación no se ha desarrollado teniendo en cuenta una serie de aspectos relacionados con la seguridad desde el inicio. Si esta tendencia continúa, los problemas de seguridad podrían ser mucho peor en el futuro. Cualquier ataque real va dirigido contra el software.

El problema de la seguridad del software no se soluciona con una receta milagrosa. Por ser un problema de múltiples facetas, requiere distintas soluciones. Mejorar la seguridad del software y salvaguardar las infraestructuras de TI es un problema de investigación y educación de las

universidades; un problema de motivación, procesos y experiencia para los fabricantes; un problema de requisitos para los clientes; un problema de pruebas y calidad para los desarrolladores; un problema de mantenimiento y aplicación de parches para la administración de TI; un problema de facilidad de uso para los usuarios y un problema de cumplimiento con la ley para los gobiernos.

McGraw en [3], plantea que la seguridad del software se basa en tres pilares fundamentales: la administración del riesgo, la aplicación de prácticas específicas en etapas del ciclo de vida de desarrollo y el conocimiento. Este tercer pilar involucra la captura, encapsulamiento y distribución del conocimiento de seguridad que puede ser usado para proveer fundamentos sólidos para prácticas de seguridad de software [3]. Por otro lado Barnun y McGraw en [5] plantean que los desarrolladores de software deben aprovechar el conocimiento y experticia adquirida en proyectos anteriores para mejorar la seguridad de los nuevos proyectos. Es en este contexto, que el presente trabajo plantea considerar algunos modelos, que si bien es cierto no han sido desarrollados específicamente para el desarrollo de software, permiten tener una visión integradora que puede ser utilizada para apoyar la realización de procesos de software que contribuyan con la fabricación y desarrollo de productos software de calidad, donde la seguridad, contribuye a lograr este objetivo. Lo que se plantea entonces, es la obtención de conocimiento de seguridad, a partir de la madurez alcanzada por el área de TI, adaptada a los procesos de software seguro.

2.1. Systems Security Engineering Capability Maturity Model (SEI)

Es un modelo de referencia para la incorporación de la Ingeniería de Seguridad en las organizaciones. Dentro del alcance de SSE-CMM, se encuentra la presentación de una serie de actividades para lograr el desarrollo de productos de software confiables, y alcanzar un ciclo de vida para sistemas seguros [6]. La propuesta se desarrolla, considerando que la ingeniería de seguridad no es una actividad que pueda desarrollarse en forma aislada de otras especialidades de la ingeniería, en especial de la ingeniería de sistemas y de software.

De todos los modelos presentados, éste es el que mayor relación y adecuación tiene respecto al desarrollo de productos de software seguros.

SSE-CMM divide la ingeniería de seguridad en tres áreas básicas: riesgo, ingeniería y aseguramiento [6]. La relación de estas áreas con el objetivo del presente artículo está dada por lo siguientes argumentos:

- Riesgo, busca identificar y priorizar los peligros asociados al desarrollo de productos o sistemas.
- Ingeniería, trabaja con otras disciplinas para implementar soluciones a los peligros identificados, en este caso, se relaciona con la Ingeniería de Software.
- Aseguramiento, tiene como objetivo certificar que las soluciones implementadas son confiables.

El modelo se estructura en dos dimensiones: Dominios y Capacidades. Un dominio es un conjunto de prácticas básicas que definen la ingeniería de seguridad, mientras que una capacidad se refiere a las prácticas genéricas que determinan la administración del proceso e institucionalizan la capacidad. Existen veintidós áreas de proceso que contienen ciento veintinueve prácticas básicas.

La siguiente tabla destaca las áreas de procesos que apoyan el desarrollo de software seguro. La información completa del modelo se encuentra en [6].

Área de Proceso	Aporte a la seguridad del software
PA03. Valoración de riesgos de seguridad	El conocimiento adquirido en esta área de proceso puede ser utilizado para el desarrollo del análisis de riesgos que pueden afectar al software, práctica recomendada para el desarrollo de productos seguros [1]
PA10. Especificar necesidades de seguridad	Esta área de proceso puede ser utilizada en la Ingeniería de Requerimientos de Seguridad, para determinar los aspectos de seguridad que deben ser incorporados al nuevo producto.
PA11. Verificación y Validación de la Seguridad.	Puede utilizarse para la generación de pruebas de seguridad [7] que deben aplicarse a los productos software.
PA17. Definir el Proceso de Ingeniería de Sistemas organizacional.	Aporta en la definición de procesos claros que incorporan la seguridad en todas las dimensiones.
PA18. Mejorar el Proceso de Ingeniería de Sistemas organizacional.	SSE-CMM define 5 niveles de madurez, por lo cual, la mejora de los procesos debe ser continua para alcanzar el siguiente nivel, lo que puede ser un aporte en la incorporación gradual de prácticas de desarrollo de software seguro.

Tabla 1. Seguridad y SSE-CMM

2.2. Control Objetives for Information and related Technology (COBIT)

El objetivo de este framework es organizar y armonizar distintos estándares internacionales, relacionados con la administración de la Tecnología de la Información en las organizaciones. Como se define en [8], COBIT presenta un conjunto de mejores prácticas, enfocadas en el control más que en la ejecución, que permiten optimizar la inversión en TI que una organización realiza. Este modelo define un conjunto de criterios de control, en base a requisitos de calidad, confianza y seguridad. El análisis e incorporación de este framework en el presente artículo, queda por completo justificado, dado que los Sistemas de Información, y sus productos software asociados, son parte de los bienes tecnológicos que una organización posee y por lo tanto, existe una relación y aporte entre la administración de la TI y las propuestas de desarrollo para software seguro.

Proceso	Aporte a la seguridad del software
PO6. Comunicar las aspiraciones y directrices de la Administración.	Dado que la incorporación de prácticas específicas para el desarrollo de software seguro debe ser un compromiso de todo el equipo, este proceso puede ayudar a que la Administración defina explícitamente políticas de seguridad y su compromiso con la calidad.
PO8. Administrar la Calidad.	Este conocimiento es un aporte, dado que la seguridad es un atributo de calidad de los productos software, por lo cual deben existir estándares de desarrollo que incorporen los objetivos de seguridad desde etapas tempranas.
PO9. Valoración y administración de riesgos.	Es un aporte para la aplicación de la práctica de análisis de riesgo, recomendada para la seguridad del software [1].

Tabla 2. Seguridad y COBIT, dominio Planear y Organizar

Este modelo se organiza en función de cuatro dominios, los que a su vez se dividen en procesos formados de actividades específicas, que definen los objetivos de control que una organización debería implementar. Dado que el objetivo del presente artículo es ver la relación existente entre los modelos y el proceso de desarrollo de software seguro, las tablas 2, 3 y 4 (Una

por cada dominio de interés) muestran los procesos específicos a considerar durante la implementación de software. En [8] se encuentra la descripción completa y detallada de todos los dominios, procesos y actividades incorporadas en COBIT.

Proceso	Aporte a la seguridad del software
AI2. Adquirir y mantener aplicaciones software.	Todo el conocimiento y experiencia de este proceso puede ser utilizado y complementado con las nuevas propuestas de desarrollo de software seguro [9], [10], [11].
AI7. Instalación y acreditación de soluciones y cambios.	Relacionado con los procedimientos de aceptación de los nuevos productos, puede aportar a explicitar las pruebas de seguridad.

Tabla 3. Seguridad y COBIT, dominio Adquirir e implementar

Proceso	Aporte a la seguridad del software
DS2. Administración de servicios prestados por terceros.	Este punto puede ser complementado con normas y controles explícitos para el momento exteriorizar el desarrollo de software.
DS5. Garantizar la seguridad de los sistemas.	Es muy importante que se declare explícitamente la preocupación por la seguridad, lo cual es un aporte a todas las nuevas propuestas de seguridad del software.
DS10. Administración de problemas.	Los problemas de fallas de seguridad de software son parte del conocimiento histórico que todos los desarrolladores de software de la organización deberían tener a su disposición, de manera tal, de evitar la repetición de errores cometidos, en este aspecto, se debe aprovechar este proceso COBIT.

Tabla 4. Seguridad y COBIT, dominio Entrega y soporte

2.3. Information Technology Infraestructure Library (ITIL)

Ofrece un marco común para todas las actividades del departamento TI, como parte de la provisión de servicios, basado en la infraestructura tecnológica. Estas actividades se dividen en procesos, que proporcionan un marco eficaz para lograr una Gestión de Servicios de Tecnologías de la Información más madura. Proporciona una descripción detallada de una serie de buenas prácticas, a través de una amplia lista de roles, tareas, procedimientos y responsabilidades que pueden adaptarse a cualquier organización de TI [12].

Librería ITIL	Aporte a la seguridad del software
Soporte del Servicio: En los aspectos relacionados con la Gestión de Incidentes y Gestión de Problemas [12]	Permiten la implementación de controles de seguridad con el fin de mantener los aspectos tecnológicos de la organización, lo cual podría ser adaptado para el desarrollo de software.
Provisión del Servicio: En cuanto a los aspectos de Gestión de Disponibilidad y Gestión de Continuidad de Servicios.[12]	Influyen de forma directa con los objetivos de seguridad, cuando se definen dentro de la organización y se asegura la demanda que requiere el negocio en cuanto a los activos tecnológicos que tiene y requiere para el desempeño de sus actividades. Esto es un aporte dado que los productos software son un activo tecnológico.

Tabla 5. Seguridad e ITIL

Los Procesos que esta librería describe, bajo los cuales muestra que son requeridos para el manejo eficiente y efectivo de la infraestructura tecnológica, existen aquellos que velan por la disponibilidad de los activos de información como requiere la organización y los demás que apoyan y facilitan la gestión de funciones y actividades de seguridad como el tratamiento

y solución de problemas, implementación de controles y políticas necesarias dentro de la organización. Por lo tanto a continuación (Ver Tabla 5) de forma general se presentan la relación entre el modelo que plantea ITIL y los aspectos que se consideran más relacionados con la seguridad, teniendo en cuenta que ITIL suministra un conjunto extenso y coherente de buenas prácticas para la dirección del servicio de informática y los procesos relacionados, promoviendo un enfoque de calidad para conseguir la eficacia de la empresa y la eficiencia en el uso de TI [13], por lo tanto dentro de dicha calidad hacia la eficacia se encuentra contemplada la seguridad.

Finalmente, resaltar la importancia que tienen los procesos que se plantean en esta metodología por medio de la cual se han incorporado conceptos de CMMI para su planteamiento y dentro de los aspectos de seguridad tiene incidencia especial ya que encamina a una organización con procesos definidos bajo parámetros de calidad.

3. DIMENSIONES DE LA SEGURIDAD

Los tres modelos anteriormente presentados, incorporan la seguridad desde distintas perspectivas, algunas de las cuales son coincidentes y otras complementarias. A partir de estas singularidades y similitudes se puede determinar una definición multidimensional de la seguridad en relación al desarrollo de software.

Se confirma y reitera lo que se plantea en las secciones previas de este trabajo, en relación a que los problemas de seguridad no se solucionan incorporando mecanismos de protección externos al producto de software, mas bien, para conseguir que un producto de software sea seguro durante su explotación y vida útil, se deben considerar durante su desarrollo las siguientes dimensiones: Proceso utilizado para el desarrollo del producto; Factores humanos relacionados con las habilidades y conocimientos que deben poseer los equipos desarrolladores; Prácticas organizacionales declaradas e instauradas que deben respetarse y aplicarse; y Productos obtenidos, para cada uno de los cuales se debe realizar una rigurosa validación y verificación que cumple con los estándares y requisitos de seguridad declarados

La figura 1 esquematiza los puntos coincidentes entre los modelos en relación al desarrollo de software seguro y las dimensiones identificadas en el párrafo anterior.

El aspecto coincidente en todos los modelos, es el planteamiento de lo importante que es que las organizaciones declaren explícitamente los requerimientos de seguridad que apoyarán el logro de sus objetivos de negocio. Esto debe ir acompañado de prácticas de seguridad bien definidas, conocidas, respetadas y aplicadas por todos los integrantes de la organización. Los modelos COBIT y SSE-CMM definen un conjunto de prácticas que deberían tenerse en consideración. Este aspecto aporta a la dimensión de Prácticas Organizacionales definida previamente.

Tanto SSE-CMM como COBIT plantean la necesidad que existan procesos de desarrollo de software que estén claramente definidos. Este aspecto ayuda a la dimensión Proceso mencionada previamente. Las organizaciones interesadas en incorporar esta dimensión, deben aplicar las propuestas existentes respecto al desarrollo de software seguro, por ejemplo, aquellas planteadas en [1].

Los modelos COBIT e ITIL, aportan a la dimensión Producto, desde la perspectiva que ambos definen cómo administrar los productos correspondientes a TI.

La dimensión de Factores Humanos, se ve beneficiada por el modelo SSE-CMM, donde se presentan una serie de procesos que deberían ser desarrollados en las organizaciones para incorporar la seguridad,

justificando la necesidad de que existan Ingenieros de Seguridad, que trabajen en conjunto a los otros dominios de la Ingeniería, en especial con los Ingenieros de Sistemas e Ingenieros de Software.
El modelo SSE-CMM, plantea la existencia de distintos niveles de madurez que puede alcanzar una organización en relación a sus prácticas de seguridad, aspecto que influye en las cuatro dimensiones de seguridad planteadas.

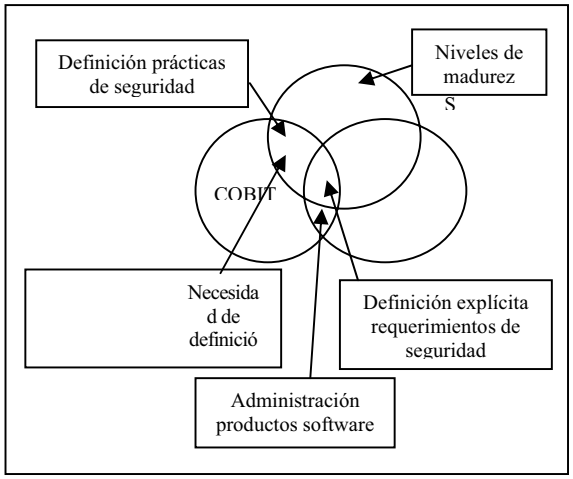

Figura 1. Aspectos coincidentes entre SSE-CMM, COBIT e ITIL en relación a la seguridad del software

4. CONCLUSIONES

Las organizaciones son cada vez más dependientes de los servicios que ofrece la informática, ya que satisfacen sus objetivos corporativos y cubren necesidades de la empresa. Esta dependencia muy lejos de decaer, día a día está en constante crecimiento, creando así una verdadera necesidad de servicios de informática de calidad para combinar con las necesidades de la empresa y los requerimientos de los diferentes usuarios cada vez que surgen.
Los modelos presentados y los procesos y/o prácticas que éstos plantean, permiten ofrecer a las organizaciones una respuesta a sus necesidades de administración y seguridad de TI, siempre enfocando el nivel de calidad que éstas requieren. Sin embargo, con toda la importancia y efectividad de los modelos analizados en cuanto a las características de calidad y sus esfuerzos por responder a este tipo de problemas en la industria del software; las necesidades a las que se da respuesta carecen de medios efectivos que apoyen los procesos que conduzcan a la obtención de productos con altos niveles de calidad en cuanto a la seguridad específicamente.
Los bienes relacionados a la Tecnología de la Información, son cada vez más valorados por las organizaciones, por lo cual, cualquier esfuerzo por mejorar su seguridad genera ganancias. En específico, las aplicaciones software y sistemas de información, deben comenzar a ser desarrollados, considerando la seguridad como una propiedad emergente, que no puede ser atacada al finalizar el ciclo de desarrollo, si no, desde las etapas tempranas. Al pensar en la seguridad, se deben incorporar los dominios planteados en la sección anterior, única forma de lograr efectivamente productos en realidad seguros, preparados para resistir los ataques a los cuales se verá expuesto.
Para conseguir este objetivo, se deben aprovechar e incorporar modelos y frameworks que considerados y aplicados en conjunto pueden ser potenciados. Por este motivo, se realizó el presente estudio, resaltando cómo los modelos presentados, pueden fortalecer los esfuerzos para el desarrollo de software seguro.
En particular, de cada modelo se puede resaltar lo siguiente: ITIL suministra un conjunto extenso y coherente de buenas prácticas para la dirección del servicio de informática y los procesos relacionados, promoviendo un enfoque de calidad para conseguir la eficacia de la empresa y la eficiencia en el uso de TI.
SSE-CMM presenta una serie de áreas de procesos que se encuentran estrechamente relacionadas con las propuestas presentadas en [1], para la mejora del ciclo de desarrollo.

COBIT, a pesar de tener un objetivo mucho más amplio que sólo el desarrollo de software, también es un excelente complemento que está en sintonía con la obtención de productos software seguro.

Finalmente, se debe resaltar la relevancia de la determinación y declaración explícita de los requerimientos de seguridad que la organización debe cumplir para alcanzar sus objetivos de negocio, sin arriesgar ni comprometer sus bienes correspondientes a TI. Esta afirmación es corroborada por los tres modelos presentados. Es así, como el área de Ingeniería de Requerimientos de Seguridad, debe asumir un papel primordial en el desarrollo de software, permitiendo la incorporación de aspectos de seguridad funcional como de seguridad como propiedad emergente del sistema en su totalidad.

5. REFERENCES

[1] McGraw, G.; 2004, Software Security,IEEE Security & Privacity, 80-83

[2] Viega J.; McGraw, G., 2001, Building Secure Software: How to avoid security problems the right way,

[3] McGraw, G., 2006, Software Security. Building Security In,

[4] Olsina, L. A.; Rossi, G. H.; Cueva, L. J., 1999, Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web, Universidad Nacional de La Plata, 257

[5] Barnum, S.; McGraw, G., 2005, Knowledge for Software Security,IEEE Security & Privacity, 74-78

[6] SSE-CMM, S.-C. Project, 2003, Model Description Document, Systems Security Engineering Capability Maturity Model, 326

[7] Potter, B. M., 2004, Software security testing, IEEE Security & Privacy Magazine, 81-85

[8] IT Governance Institute, Cobit 4.0 The Newest Evolution Of Control Objectives For Information And Related Technology, The World's Leading It Control And Governance Framework

[9] Jaferian, P.; Elahi, G.; Ayatollahzadeh, M.; Sadeghian, B., 2005, RUPSec: Extending Business Modeling and Requirements Disciplines of RUP for Developing Secure Systems

[10] Apvrille, A.; Pourzandi, M., 2005, Secure Software Development by Example,IEEE Security & Privacity, 10-17

[11] Viega, J., 2006, Security in the software development lifecycle. An introduction to CLASP, the Comprehensive Lightweight Application Security Process,

[12] Kemmerling, G.; Pondman, D., 2004, Gestión de Servicios TI, una introducción a ITIL,O. o. G. C. I. S. Sipport

[13] Hochstein, A. Z.; Brenner, R., 2005, ITIL as common practice reference model for IT service management: formal assessment and implications for practice, 704-710