Seguridad y Calidad en Aplicaciones Web

Unidad N° 6: Normas

Referente de Cátedra: Walter R. Ureta
Plantel Docente: Cintia Gioia, Juan Monteagudo, Walter R. Ureta

CMM

El Modelo de Madurez de Capacidades o CMM (Capability Maturity Model), es un modelo de evaluación de los procesos de una organización, predecesor de CMMI. Fue desarrollado inicialmente para los procesos relativos al desarrollo e implementación de software por la Universidad Carnegie-Mellon para el SEI (Software Engineering Institute).

CMMi

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 de producto con miembros procedentes de la industria, del gobierno y del Software Engineering Institute (SEI).

SSE-CMM

El System Security Engineering Capability Maturity Model o Modelo de Madurez de Capacidades en la Ingeniería de Seguridad de Sistemas es un modelo derivado del CMM y que describe las características esenciales de los procesos que deben existir en una organización para asegurar una buena seguridad de sistemas.

Nació a partir de 1993 bajo los auspicios de la Agencia Nacional de Seguridad (NSA) de los E.U.A., con la participación de numerosas compañías de los sectores de tecnologías de la información, seguridad y defensa

CMM-Niveles

Capability Level 1 – Performed Informally
Capability Level 2 – Planned and Tracked
Capability Level 3 – Well Defined
Capability Level 4 – Quantitatively Controlled
Capability Level 5 – Continuously Improving

CMM-SECURITY BASE PRACTICES

PA01 – Administer Security Controls
PA02 – Assess Impact
PA03 – Assess Security Risk
PA04 – Assess Threat
PA05 – Assess Vulnerability
PA06 – Build Assurance Argument
PA07 – Coordinate Security
PA08 – Monitor Security Posture
PA09 – Provide Security Input
PA10 – Specify Security Needs
PA11 – Verify and Validate Security

CMM-PROJECT AND ORGANIZATIONAL BASE PRACTICES

PA12 – Ensure Quality
PA13 – Manage Configurations
PA14 – Manage Project Risk
PA15 – Monitor and Control Technical Effort
PA16 – Plan Technical Effort
PA17 – Define Organization's Systems Engineering Process
PA18 – Improve Organization's Systems Engineering Processes
PA19 – Manage Product Line Evolution
PA20 – Manage Systems Engineering Support Environment
PA21 – Provide Ongoing Skills and Knowledge
PA22 – Coordinate with Suppliers

SSE-CMM-Dimensiones

Common Features	Process Areas: PA01 – Administer Security Controls	PA02 – Assess Impact	PA03 – Assess Security Risk	PA04 – Assess Threat	PA05 – Assess Vulnerability	PA06 – Build Assurance Argument	PA07 – Coordinate Security	PA08 – Monitor Security Posture	PA09 – Provide Security Input	PA10 – Specify Security Needs	PA11 – Verify and Validate Security	PA12 – Ensure Quality	PA13 – Manage Configuration	PA14 – Manage Project Risk	PA15 – Monitor and Control Technical Effort	PA16 – Plan Technical Effort	PA17 – Define Org. Systems Eng. Process	PA18 – Improve Org. Systems Eng. Process	PA19 – Manage Product Line Evolution	PA20 – Manage Systems Eng. Support Env.	PA21 – Provide Ongoing Skills and Knldge	PA22 – Coordinate with Suppliers
5.2 Improving Proc. Effectiveness																						
5.1 Improving Org. Capability																						
4.2 Objectively Managing Perf.																						
4.1 Establish Meas. Quality Goals																						
3.3 Coordinate Practices																						
3.2 Perform the Defined Process																						
3.1 Defining a Standard Process																						
2.4 Tracking Performance																						
2.3 Verifying Performance																						
2.2 Disciplined Performance																						
2.1 Planned Performance																						
1.1 Base Practices Are Performed																						
	Security Engineering Process Areas											Project and Organizational Process Areas										

SAMM - Software Assurance Maturity Model

El modelo de madurez para el aseguramiento de software es un marco de trabajo abierto para ayudar a las organizaciones a formular e implementar una estrategia de seguridad para Software que sea adecuada a las necesidades específicas que está enfrentado la organización.

OWASP
The Open Web Application Security Project

SAMM - Software Assurance Maturity Model

Las bases de este modelo están construidas alrededor de las funciones de negocio relacionadas al desarrollo de Software, se incluyen una serie de prácticas relacionadas a cada función.

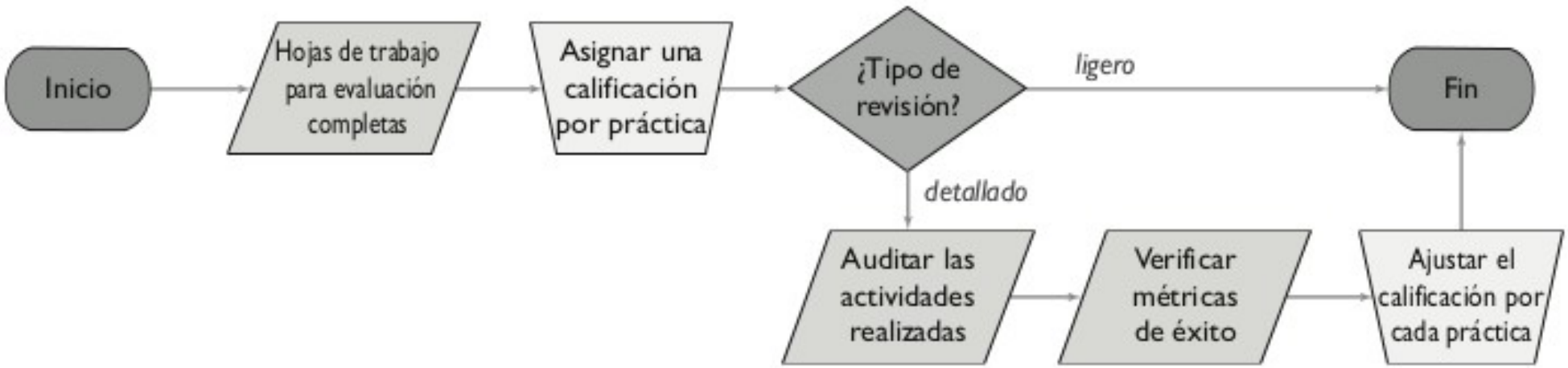
SAMM - Software Assurance Maturity Model

Niveles de Madurez

Cada una de las prácticas de seguridad tiene tres niveles de madurez bien definidos y un nivel inicial (cero) implícito. Los detalles de cada nivel difieren entre las prácticas pero generalmente representan:

- Punto de inicio implícito, las actividades en la practica no se han realizado.
- Entendimiento inicial y provisión ad hoc de la práctica de seguridad.
- Incremento en la eficiencia y/o efectividad de la práctica de seguridad.
- Dominio amplio de la práctica de seguridad.

SAMM - Revisiones



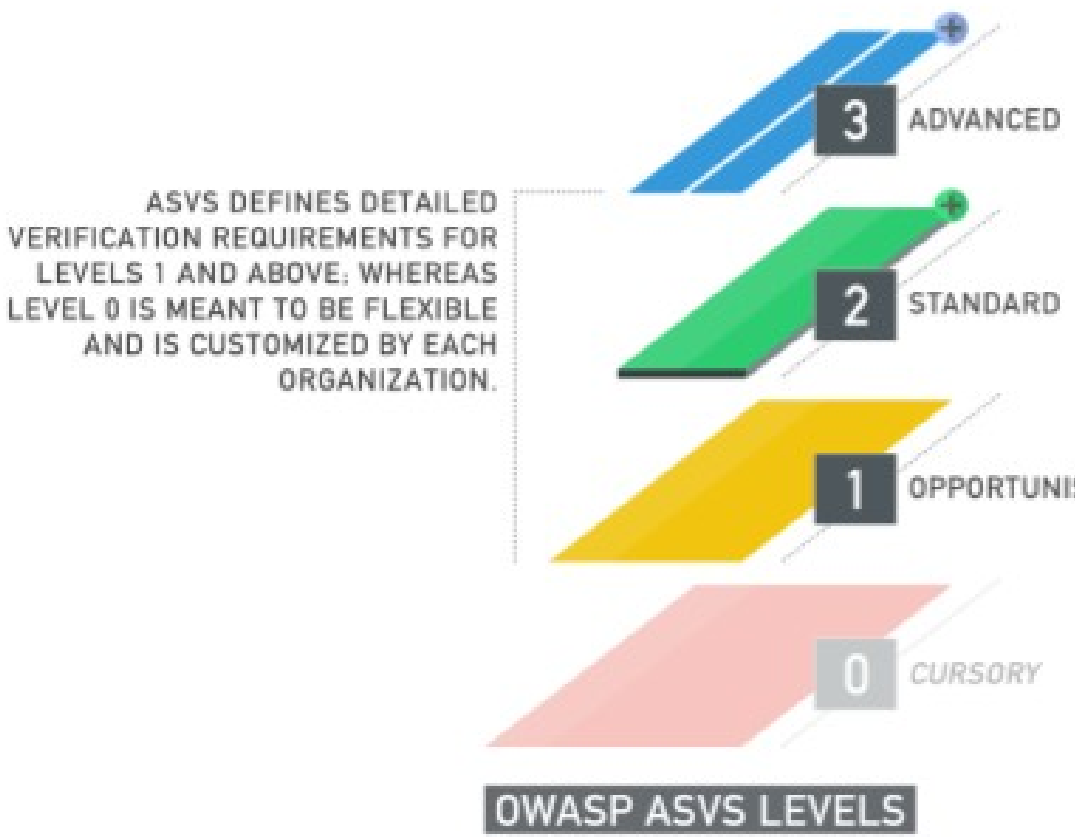
Después de responder las preguntas de las hojas de trabajo, evalúe la columna de respuestas para determinar el nivel. El cual se indicaría con respuestas afirmativas para todas las preguntas sobre los marcadores a la derecha de la columna de respuestas.

ASVS – Application Security Verification Standard

El objetivo principal del ***Application Security Verification Standard (ASVS)*** del OWASP es normalizar el rango de cobertura y el nivel de rigurosidad disponible en el mercado cuando se realiza la verificación de seguridad de aplicaciones web.

Este estándar podrá ser utilizado tanto por los consumidores como por los proveedores del servicio o la herramienta.

ASVS – Application Security Verification Standard



Indica que se defiende adecuadamente contra todas las vulnerabilidades de seguridad avanzadas, y también demuestra principios de un buen diseño de seguridad.

Indica que se defiende adecuadamente contra las vulnerabilidades de seguridad frecuentes cuya existencia plantea un riesgo de moderado a grave.

Indica que se defiende adecuadamente contra vulnerabilidades de seguridad fáciles de descubrir.

Certificación opcional, indica que la aplicación ha pasado algún tipo de verificación definida por la organización.

ASVS – Áreas de Requerimientos de Seguridad

V2. Authentication
V3. Session Management
V4. Access Control
V5. Malicious Input Handling
V7. Cryptography at Rest
V8. Error Handling and Logging
V9. Data Protection
V10. Communications
V11. HTTP
V13. Malicious Controls
V15. Business Logic
V16. File and Resource
V17. Mobile

Normas

- **ISO 9001** en el alcance sobre el software y sobre los procesos productivos de la organización. No siempre sobre el desarrollo, puede ser en la identificación de requisitos, en el propio desarrollo y por ejemplo en la entrega y mantenimiento.
- **ISO/IEC 9003** Ingeniería del software. Guía de aplicación de la ISO 9001:2000 al software (NO es CERTIFICABLE. Es una norma de buenas prácticas para definir con más detalle los conceptos de software sobre los procesos de la organización).
- **ISO/IEC 12207** I*nformation Technology / Software Life Cycle Processes*, es el estándar para los procesos de ciclo de vida del software de la organización. Es la base para ISO 15504-SPICE.

Normas

- **ISO/IEC 15504** (conocida como SPICE - *Software Process Improvement And Assurance Standards Capability Determination*). Un conjunto de 7 normas para establecer y mejorar la capacidad y madurez de los procesos de las organizaciones, proporcionando los principios requeridos para realizar una **evaluación de la calidad de los procesos.** La definición de los procesos se realiza sobre ISO/IEC 12207. La familia de normas 15504 espera que la nueva **ISO 29110** sea publicada para crear definitivamente el esquema internacional de certificación, que actualmente está creado con procesos de calidad en las entidades de certificación (realizando evaluaciones externas sobre I**SO/IEC 15504-2** e **ISO/IEC TR 15504-7:2008**.

- **ISO/IEC 9126.** Desarrolladas entre 1991 y 2001. *Software engineering – Product quality* consta de 4 partes. La serie de normas ISO/IEC 9126 define las características de calidad del **producto de software** (parte 1), las métricas internas y externas (partes 2 y 3), y la calidad en uso, que explica cómo la calidad del producto está sujeta a las condiciones particulares de uso (parte 4).

Normas

- **ISO/IEC 14598.** Desarrolladas entre 1999 y 2001. *Software product evaluation,* **Evaluación del producto de software**, la familia consta de 6 partes. Directamente relacionada con ISO 9126.
- **ISO 25000.** La familia de normas 25000 establecen un modelo de calidad para el **producto software** además de definir la evaluación de la calidad del producto. Tiene 5 partes publicadas. Pretenden sustituir a ISO 9126 e ISO 14598 ya que desde 2001 no se publicaron nuevas versiones.
- **SCRUM**. No es una **norma**, es un método sencillo y práctico para empezar a practicar calidad. Fabricar y gestionar el desarrollo en tres fases fundamentales: una breve fase de planificación, en la cual se realizan las labores básicas de una planificación breve: visión general del proyecto (estimación muy general, viabilidad del sistema) y construcción del Backlog. por un lado y por otro el desarrollo de la arquitectura al detalle; otra de desarrollo, en la cual tienen lugar los famosos Sprints, y otra final de entrega y balance de los éxitos y fracasos logrados

Normas

- **ISO/IEC 27000:** es un conjunto de estándares desarrollados por ISO (International Organization for Standardization) e IEC (International Electrotechnical Commission), que proporcionan un marco de gestión de la seguridad de la información utilizable por cualquier tipo de organización, pública o privada, grande o pequeña.