

OWASP LatamTour
Chile 2016

Desarrollo Seguro

¿Y esto cómo me ayuda a cumplir PCI-DSS?

Carlos Allendes
Presidente Owasp Chile

 OWASP
The Open Web Application Security Project



OWASP LatamTour
Honduras 2016

Desarrollo Seguro

¿Y esto cómo me ayuda a cumplir PCI-DSS?



Antecedentes del Expositor

OWASP
The Open Web Application Security Project

Carlos Allendes Drogueut (carlos.allendes@owasp.org)

- Ingeniero Civil en Informática, USACH
- Presidente capítulo chileno OWASP
- Co-fundador capítulo OWASP Honduras y Rep.Dominicana
- Owner en www.qualityfactory.cl (callendes@qualityfactory.cl)
- Asociado en www.dataactiva.cl (callendes@dataactiva.cl)
- Experiencia y proyectos
 - CMMi, AGILE, Ingeniería de Software aplicada.
 - QA y Testing, Testing como servicio externalizado.
 - PCI DSS, acreditación en seguridad.
 - ITIL, implantación de procesos.

Agenda

OWASP
The Open Web Application Security Project

- Presentación del expositor
- **La deuda técnica (el problema)**
- Ingeniería de Software (la solución)
- El modelo CMMi
- El modelo SAMM
- Aplicación de SAMM en el mundo real

**La deuda técnica
(el problema)**

OWASP
The Open Web Application Security Project

"Deuda Técnica" es una expresión conceptual, aplicada al impacto económico debido a la mala calidad del software (re-trabajo, atrasos, mayor uso de recursos).

Categoría	Porcentaje
Fracaso	52%
Éxito	48%
Rehecho o Abandonado	19%
Nunca Usado	3%
Usado Sin Modificación	2%
Usado con Cambios	46%
Pagado pero Nunca Liberado	30%

Case Strategies, jul. 2010

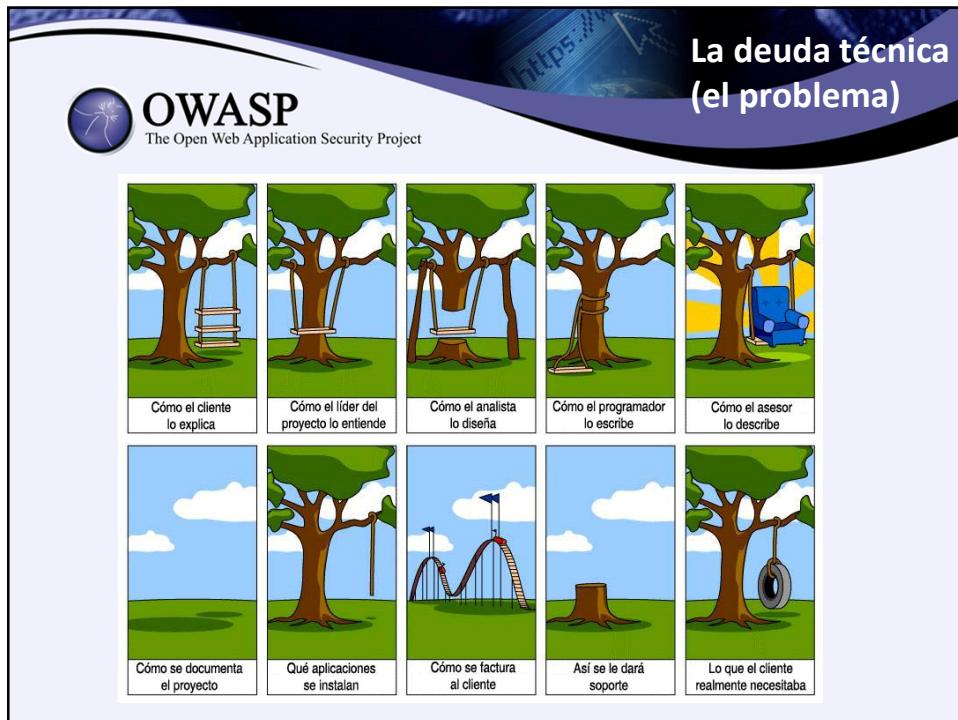
**La deuda técnica
(el problema)**

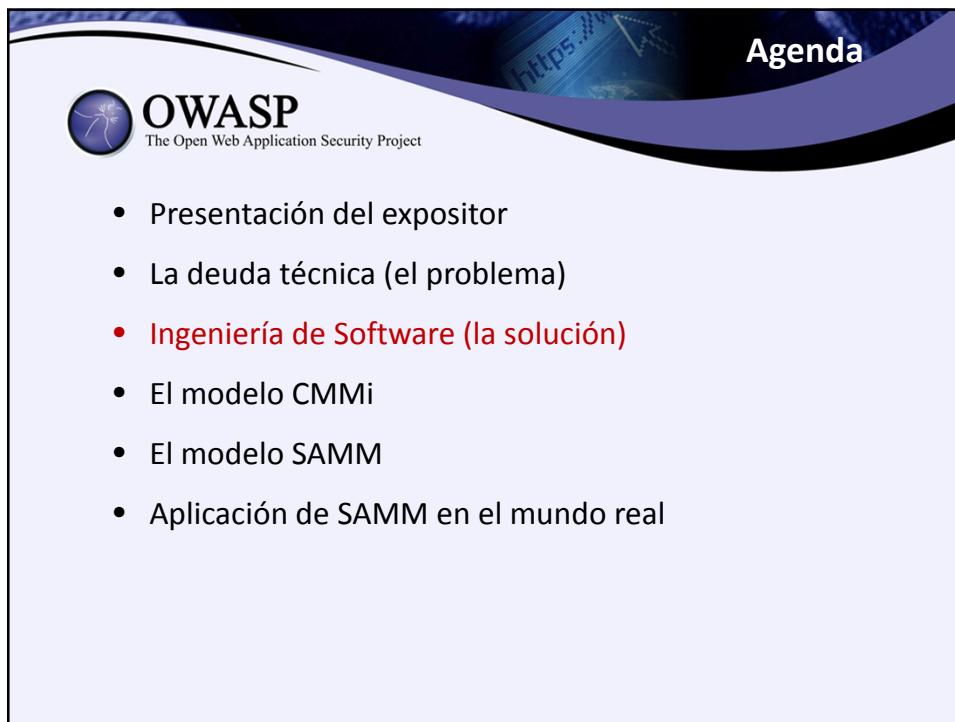
OWASP
The Open Web Application Security Project

El Desarrollo de Software es una ARTESANIA...

...y en las artes, el producto final depende del artista.

Y en Desarrollo de Software abundan los artistas...





The slide features the OWASP logo at the top left. To the right, the word "Agenda" is displayed in white text on a dark blue banner. The main content area is a light purple rectangle containing a bulleted list of topics:

- Presentación del expositor
- La deuda técnica (el problema)
- **Ingeniería de Software (la solución)**
- El modelo CMMi
- El modelo SAMM
- Aplicación de SAMM en el mundo real



The slide features the OWASP logo at the top left. To the right, the text "Ingeniería de Software (la solución)" is displayed in white on a dark blue banner. The main content area is a light purple rectangle containing the following text:

La ingeniería de Software.
"Aplicación de un enfoque sistemático, disciplinado y medible al desarrollo, operación y mantenimiento del software".
[IEEE, 1993]

A small image of a 3D-style maze with a red path line showing the route from the start to the end.

Ingeniería de Software (la solución)

Capas de la Ingeniería de Software.

The diagram illustrates the four layers of software engineering as stacked horizontal bars. From top to bottom, they are labeled: HERRAMIENTAS (Tools), METODOS (Methods), PROCESO (Process), and ENFOQUE DE CALIDAD (Quality Focus). A large orange arrow points upwards from the bottom layer towards the top layer, indicating a flow or dependency. To the left of the diagram, there are logos for CMMI, ISO/IEC 14598, and ISO 9000-3.

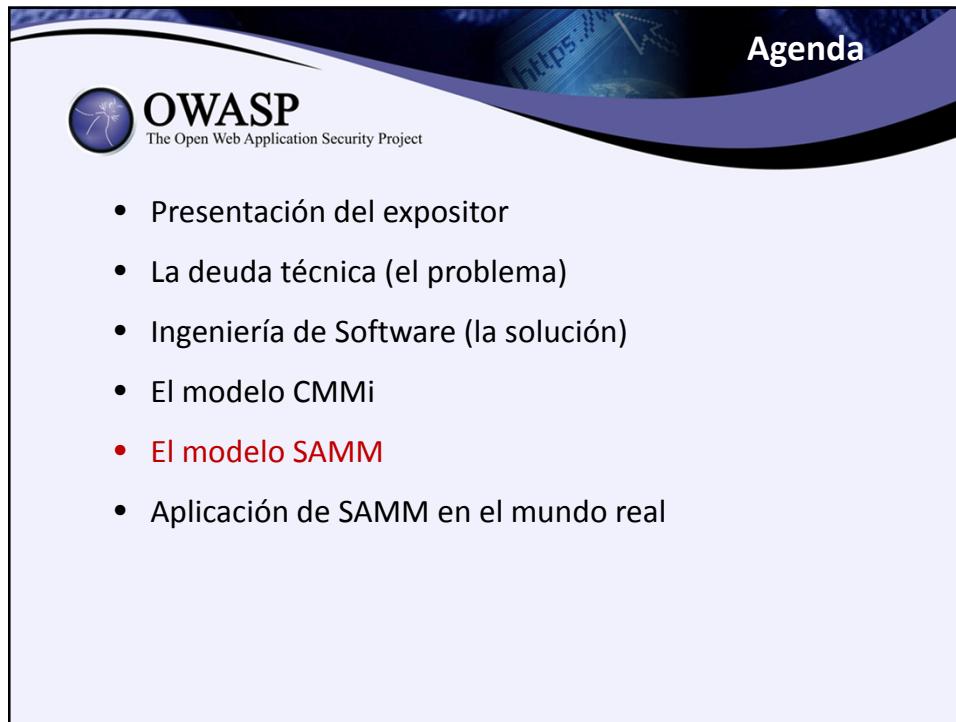
Fig. 2.1. Capas de la ingeniería del software

Agenda

OWASP
The Open Web Application Security Project

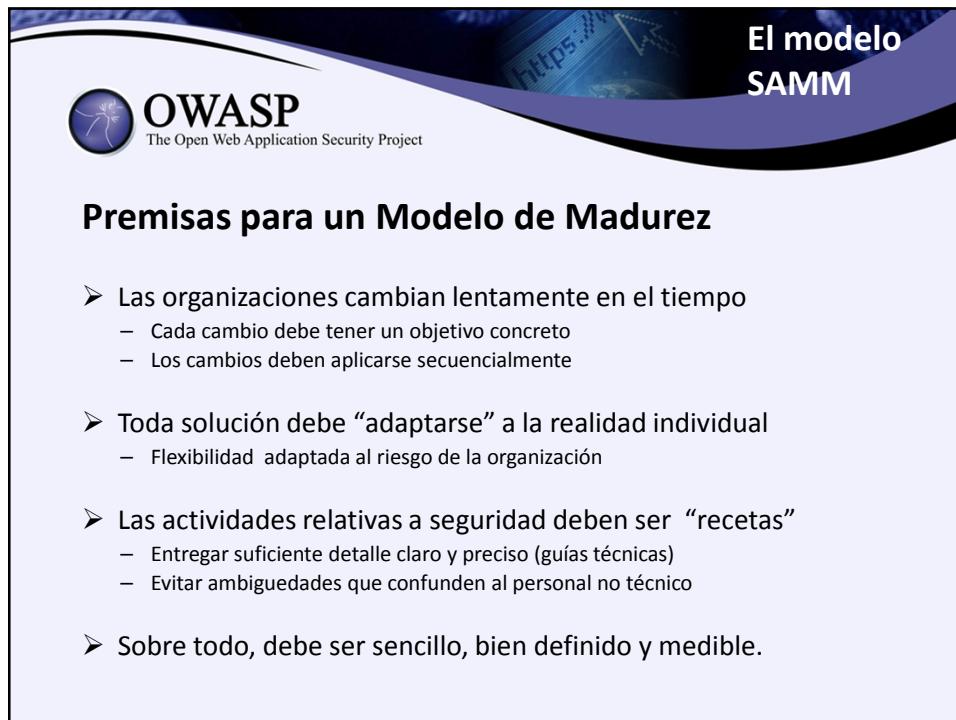
- Presentación del expositor
- La deuda técnica (el problema)
- Ingeniería de Software (la solución)
- El modelo CMMi
- El modelo SAMM
- Aplicación de SAMM en el mundo real





The slide features the OWASP logo at the top left, followed by the word "Agenda" in a large, bold, white font. The background is a dark blue gradient with a faint watermark of a computer screen showing a URL like "https://".

- Presentación del expositor
- La deuda técnica (el problema)
- Ingeniería de Software (la solución)
- El modelo CMMi
- **El modelo SAMM**
- Aplicación de SAMM en el mundo real



The slide features the OWASP logo at the top left, followed by the title "El modelo SAMM" in a large, bold, white font. The background is a dark blue gradient with a faint watermark of a computer screen showing a URL like "https://".

Premisas para un Modelo de Madurez

- Las organizaciones cambian lentamente en el tiempo
 - Cada cambio debe tener un objetivo concreto
 - Los cambios deben aplicarse secuencialmente
- Toda solución debe “adaptarse” a la realidad individual
 - Flexibilidad adaptada al riesgo de la organización
- Las actividades relativas a seguridad deben ser “recetas”
 - Entregar suficiente detalle claro y preciso (guías técnicas)
 - Evitar ambigüedades que confunden al personal no técnico
- Sobre todo, debe ser sencillo, bien definido y medible.

El modelo SAMM

OWASP
The Open Web Application Security Project

Historia de SAMM

- Versión Beta liberada en Agosto de 2008
- Creada originalmente por Fortify (ahora HP)
- Autores aún involucrados activamente
- Publicada bajo licencia Creative Commons
- Donada al proyecto OWASP
- Cambia su nombre a OpenSAMM

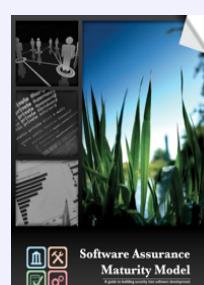


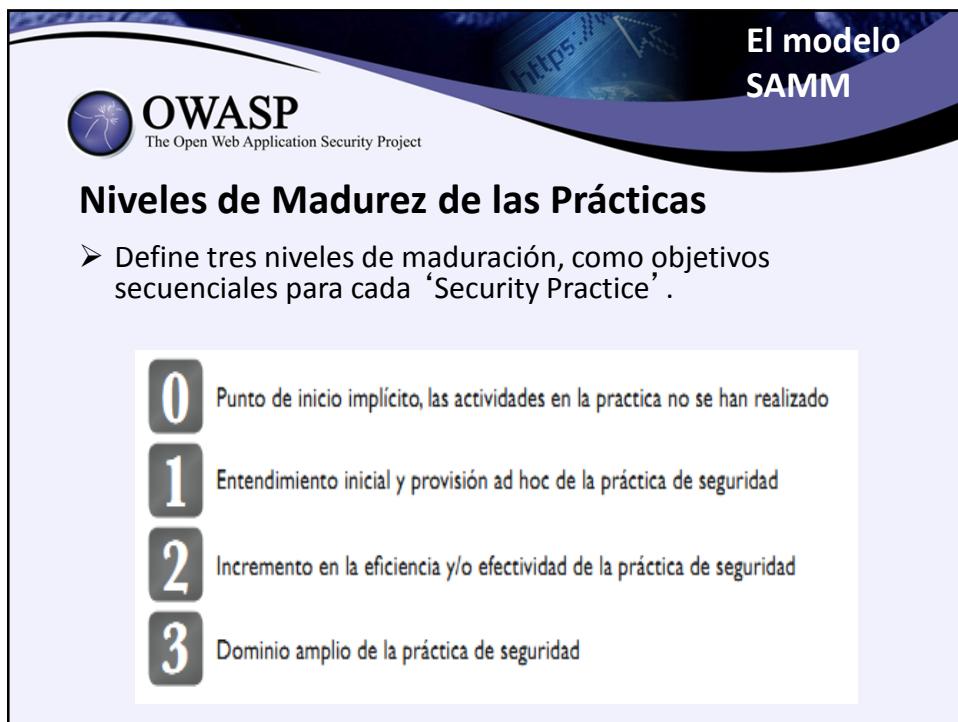
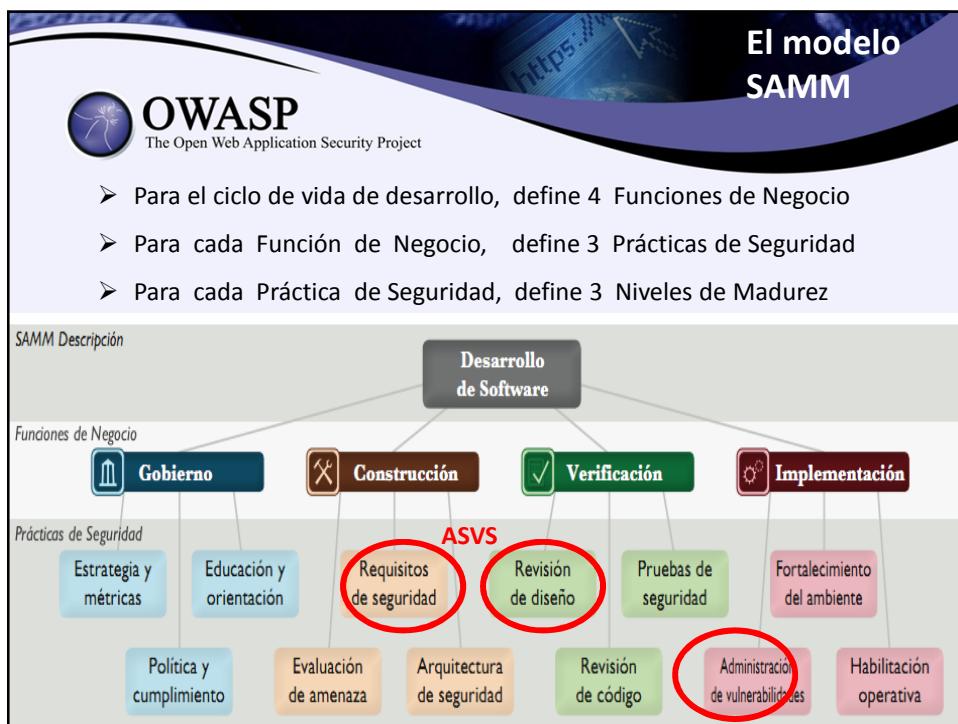
El modelo SAMM

OWASP
The Open Web Application Security Project

Utilidad de SAMM

- Metodología que sirve para evaluar las prácticas de desarrollo seguro en una organización.
- Sirve para implementar un programa de "Seguridad de aplicativos" en forma iterativa e incremental.
- Muestra objetivamente los avances en el programa de mejoras de seguridad de aplicaciones.
- Define y mide actividades relacionadas a la seguridad en toda la organización.





Ejemplo aplicado
Definir Objetivo

OWASP
The Open Web Application Security Project

>> Función de Negocio: Verificación (Testing)
>> Práctica Segura: Revisar el Código

Verificación

Resumen de actividades

Revisión de código			<i>...continúa en página 62</i>
CR 1	CR 2	CR 3	
OBJETIVOS	Encontrar oportunamente vulnerabilidades básicas a nivel de código y otros problemas de seguridad de alto riesgo	Hacer revisiones de código más precisas y eficientes durante el desarrollo a través de la automatización	Exigir un proceso de revisión de código integral para descubrir riesgos específicos de la aplicación y a nivel del lenguaje
ACTIVIDADES	A. Crear listas de verificación para la revisión de los requisitos de seguridad conocidos B. Realizar revisiones en código de puntos de alto riesgo	A. Utilizar herramientas automatizadas de análisis de código B. Integrar análisis de código en el proceso de desarrollo	A. Personalizar el análisis de código para las preocupaciones específicas de la aplicación B. Establecer puntos de control para la liberación de las revisiones de código

Ejemplo aplicado
Evaluar situación actual

OWASP
The Open Web Application Security Project

>> Checklist para evaluación del GAP

Verificación

Hoja de trabajo para evaluación

Revisión de código	Sí/No
♦ ¿La mayoría de los equipos de proyecto tienen listas de verificación basadas en los problemas más comunes?	
♦ Los equipos de proyecto ¿Generalmente realizan revisiones de algunos de los mayores riesgos en el código?	CR 1
♦ ¿Pueden la mayoría de los equipos de proyecto acceder a herramientas automatizadas de análisis de código para encontrar problemas de seguridad?	
♦ ¿La mayoría de los interesados requieren y revisan constantemente los resultados de las revisiones de código?	CR 2
♦ ¿La mayoría de los equipos de proyecto utilizan automatización para comprobar código contra los estándares de programación específicos de la aplicación?	
♦ ¿Las auditorías de rutina del proyecto necesitan lineamientos para los resultados de la revisión de código antes de la liberación?	CR 3

Ejemplo aplicado
Evaluar situación actual

➤ Realizar Evaluación (GAP análisis)

➤ SAMM aporta documentos de evaluación para cada “Práctica de Seguridad”.

** Recuerde adaptarlos a su realidad**

Práctica de Seguridad	Evaluación
Evaluación de amenaza	1
Requisitos de seguridad	0+
Arquitectura de seguridad	0
Revisión de diseño	1
Revisión de código	3
Pruebas de seguridad	1
Administración de vulnerabilidades	0+

Ejemplo aplicado
Objetivos detallados por nivel madurez

Por cada Nivel madurez, SAMM define...

- Objetivo
- Actividades
- Resultados
- Umbrales de satisfacción
- Costos
- Personal
- Niveles relacionados

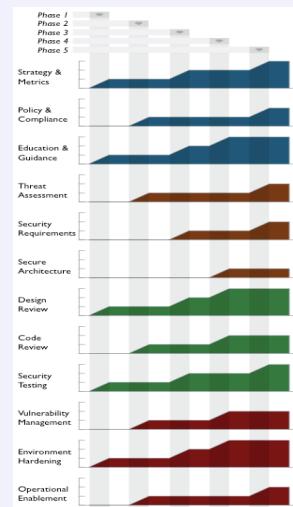
Revisión de código

Objetivos	cr 1	cr 2	cr 3
Encontrar oportunamente vulnerabilidades básicas a nivel de código y otros problemas de seguridad de alto riesgo	✓	✓	✓
Actividades	A. Crear listas de verificación para la revisión de los requisitos de seguridad conocidos B. Hacer revisiones en código a partir de alto riesgo	A. Utilizar herramientas automatizadas de análisis de código B. Integrar análisis de código en el proceso de desarrollo	A. Personalizar el análisis de código para las preoccupaciones específicas de la aplicación B. Establecer puntos de control para la liberación de las revisiones de código
Evaluación	<ul style="list-style-type: none"> • La mayoría de los equipos de proyecto crean listas de verificación basadas en los problemas más comunes? • Los equipos de proyecto ¿Generalmente realizar revisiones de seguimiento de los mejores riesgos en el código? 	<ul style="list-style-type: none"> • Pueden la mayoría de los equipos de proyecto acceder a herramientas automatizadas de análisis de código para encontrar problemas de código? • ¿La mayoría de los interesados requieren y revisan constantemente los resultados de las revisiones de código? 	<ul style="list-style-type: none"> • La mayoría de los equipos de proyecto tienen una automatización para comprobar código contra los criterios de calidad y configuración específicas de la aplicación? • Los equipos de proyecto del proyecto necesitan herramientas para los resultados de las revisiones de código antes de la liberación?
Resultados	<ul style="list-style-type: none"> • Inspección de las vulnerabilidades de código comunes que conducirían a un probable desastre de seguridad? • Revisión ligera de errores de codificación que conducen a la ejecución de software severo a la seguridad? • Obediente básica a nivel de código para el aseguramiento de la seguridad? 	<ul style="list-style-type: none"> • El desarrollo permite consideradamente auto identificar las vulnerabilidades de seguridad de código? • Resultados de análisis de código para compilar datos históricos de vulnerabilidad y generación segura por equipo? • Los interesados están conscientes de las revisiones de código no integradas para apoyar el mejor análisis de negociación? 	<ul style="list-style-type: none"> • Incrementar la confianza en la precisión y aplicabilidad de los resultados de análisis de código? • Lineamientos organizacionales para las expectativas de gestión de riesgos? • Equipos de proyecto con un objetivo a alcanzar para juzgar la seguridad a nivel de código?

Ejemplo aplicado
Plantillas de roadmap
por tipo de Industria

Plantillas de Planes de Mejora (Roadmap)

- SAMM entrega Plantillas de Planes de mejora (*Roadmaps*) para diferentes tipos de Organización (industria)
- Desarrolladores de Software Independientes
 - Organizaciones de servicios financieros (FSO)
 - Administraciones Públicas (AAPP)
- Organizaciones tipo se han elegido porque:
 - Representan los casos de uso más comunes
 - La definición de un “Plan de mejora de la seguridad” optimizado.... es diferente en cada caso.



Caso de Exito

CASOS DE EXITO



Caso de Exito

OWASP
The Open Web Application Security Project

Práctica: Revisión de Código

Resultados

CANTIDAD DE HALLAZGOS DETECTADOS (SEGURIDAD DE CÓDIGO)

Año	Mantenimientos	Proyectos	Total
2009	12.215	3.220	16.135
2010	4.757	2.237	6.994
2011	2.630	1.061	3.691
2012	105	316	422

ISO 9001:2008 PCI Data Security Standard CMMI

Caso de Exito

OWASP
The Open Web Application Security Project

Práctica: Revisión de Código

- **3,5 años**
- **2,3 millones de líneas de código**
- **96% Mejora, por reducir Hallazgos de Seguridad**

AÑO >>	año-1	año-2	año-3	año-4	
KLOCs	970	897	357	130	2.354
Security Findings	16.135	6.994	3.691	723	27.543
% Tasa de Mejora	0%	57%	77%	96%	
% Hallazgo Residual	100%	43%	23%	4%	

Caso de Exito

OWASP
The Open Web Application Security Project

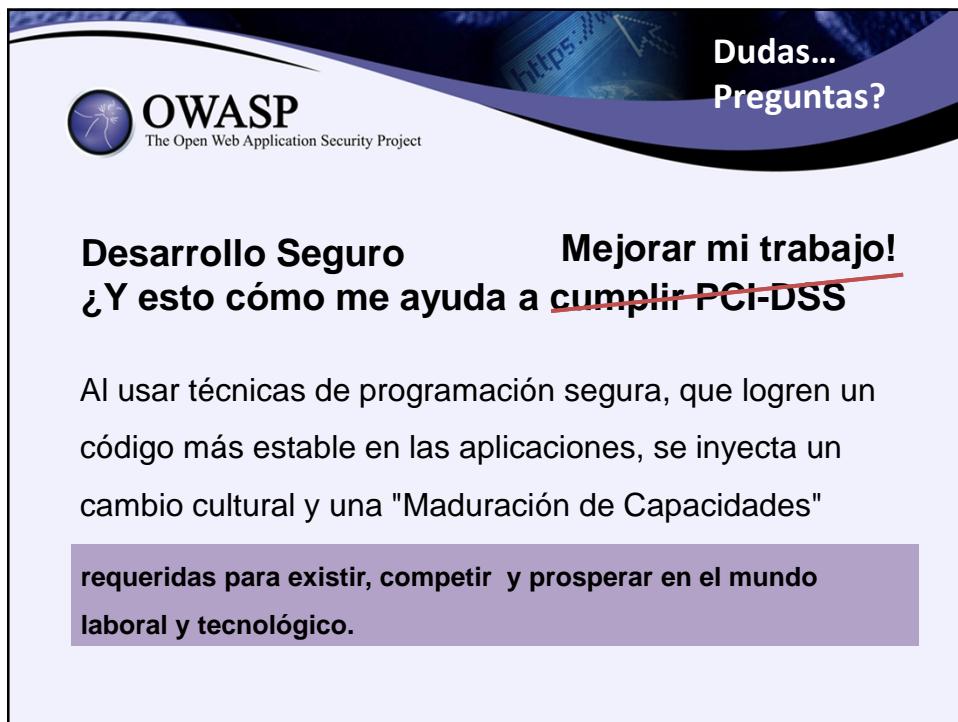
- Otros Casos de exito.. disponibles en diversos referentes de la industria TI:

Repasso...

OWASP
The Open Web Application Security Project

Pasos metodológicos... para implantar DESARROLLO SEGURO

- Evaluar las prácticas de seguridad existentes en la organización.
- Definir un plan ad-hoc de mejora en la seguridad del software basado en iteraciones bien definidas.
- Cuantificar mejoras concretas durante la aplicación del plan de mejora en la seguridad.
- Definir y medir actividades relacionadas con la seguridad en una organización.



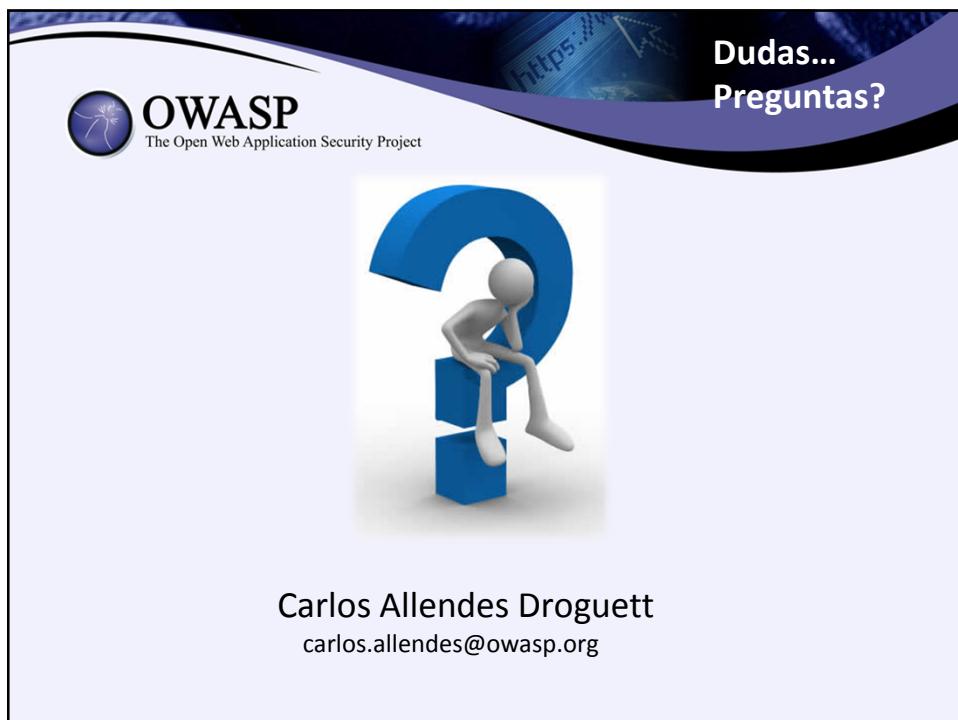
The slide features the OWASP logo and tagline "The Open Web Application Security Project". In the top right corner, there is a purple banner with the text "Dudas..." and "Preguntas?". The main content area contains two sections: "Desarrollo Seguro" and "Mejorar mi trabajo!". Below these, a question is posed: "¿Y esto cómo me ayuda a cumplir PCI-DSS". A large paragraph follows, explaining the benefits of secure programming: "Al usar técnicas de programación segura, que logren un código más estable en las aplicaciones, se inyecta un cambio cultural y una "Maduración de Capacidades" requeridas para existir, competir y prosperar en el mundo laboral y tecnológico." A large blue question mark icon is positioned in the center of the slide.

Dudas...
Preguntas?

Desarrollo Seguro **Mejorar mi trabajo!**

¿Y esto cómo me ayuda a cumplir PCI-DSS

Al usar técnicas de programación segura, que logren un código más estable en las aplicaciones, se inyecta un cambio cultural y una "Maduración de Capacidades" requeridas para existir, competir y prosperar en el mundo laboral y tecnológico.



The slide features the OWASP logo and tagline "The Open Web Application Security Project". In the top right corner, there is a purple banner with the text "Dudas..." and "Preguntas?". The central focus is a large blue question mark icon containing a small white 3D human figure sitting on one of its points. At the bottom of the slide, the name "Carlos Allendes Drogue" and email "carlos.allendes@owasp.org" are displayed.

Dudas...
Preguntas?

Carlos Allendes Drogue
carlos.allendes@owasp.org