Presentación del OWASP Testing Project

Madrid, junio 2006

El conferenciante

Javier Fernández-Sanguino <jfernandez@germinus.com>

Consultor y jefe de proyecto en la división de seguridad IT de Germinus XXI, S.A.

Ingeniero de Telecomunicación por la ETSIT-UPM,

Miembro de diversos grupos de desarrollo de software libre dentro del **proyecto Debian**, y de los de diversas herramientas de seguridad, entre otras: **Tiger**, **Nessus**, y **Bastille**.

Miembro fundador del grupo español de la **Honeynet Alliance**

Miembro del grupo **OWASP Testing**

Presentación del Proyecto OWASP Testing

-Pruebas de intrusión en aplicaciones

-Introducción al Proyecto OWASP Testing

- *Enfoque*
- *Historia*
- *Otros proyectos relacionados*

-Orientación

-Desarrollos a la fecha

-Cómo contribuir

-Referencias útiles

germinus

Pruebas de intrusión en aplicaciones

-Es la comprobación de seguridad que se suele contratar.

-Se realiza sobre aplicaciones ya desplegadas (en producción).

-No es lo más apropiado para encontrar todos los problemas:

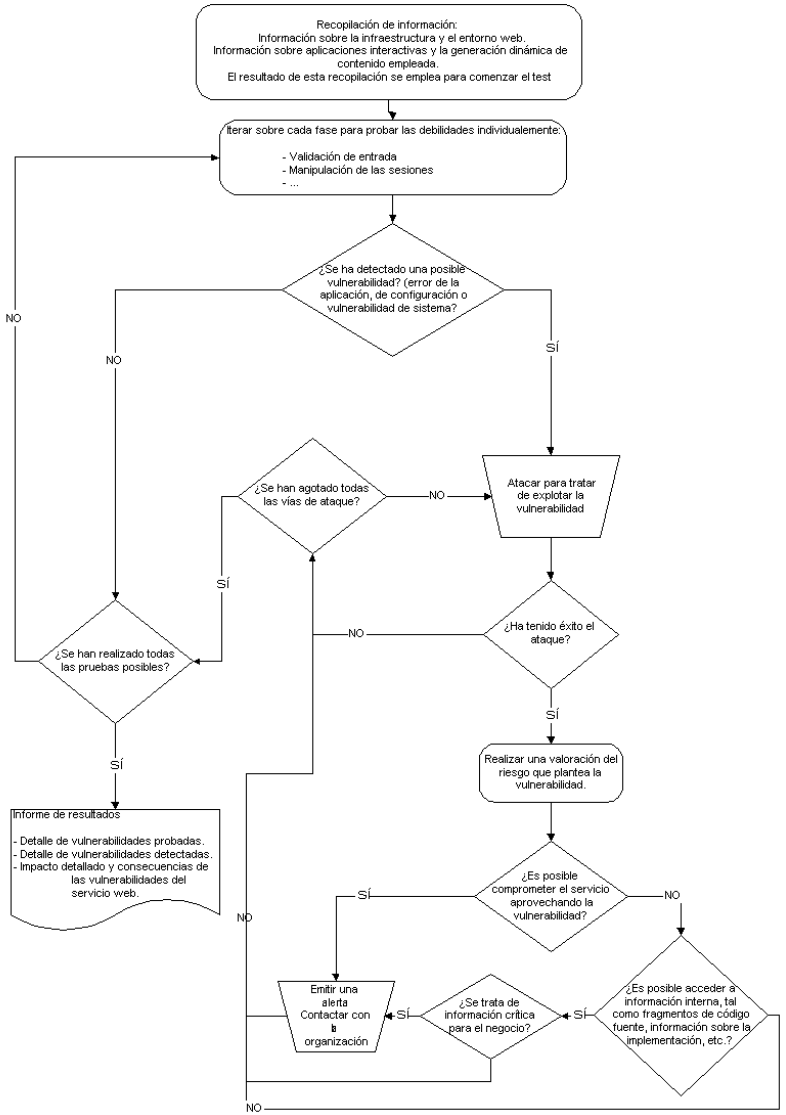
-Pero se pueden encontrar los errores de mayor envergadura.

-Se pueden detectar fallos “de concepto”.

-Existen herramientas que las automatizan

-Ayudan en tareas automáticas.

-Pero debido a la variabilidad sigue siendo algo que una persona con conocimientos hará mucho mejor.



germinus

OWASP Testing

Proyecto que tiene como objetivo crear un marco de trabajo para el desarrollo de pruebas de seguridad en aplicaciones web libremente accesible (GFDL) para profesionales y responsables de seguridad.

Resultados a la fecha:

- Guía de pruebas de seguridad en aplicaciones web.
- Documentos para asistir a la realización de pruebas (*checklists*).

Complementa a (y complementado con):

- Guía de diseño de aplicaciones web.
- Herramientas de pruebas de seguridad en aplicaciones web (Webscarab).
- OWASP Legal.

OWASP Testing

No enfocado solamente a las pruebas de intrusión sino al ámbito del ciclo de vida de desarrollo de software e introduciendo actividades de pruebas como la definición de modelos de riesgo, la revisión de código fuente y las pruebas de intrusión.

Objetivos:

-Ayudar a aquellas personas que tienen que hacer pruebas de seguridad.

-Ayudar a los a sus clientes (especificación de pruebas a realizar).

-Satisfacer la función de revisión de aplicaciones (indefinida).

-Dar un acercamiento más estructurado (evolución).

-Lograr la máxima difusión (internacional).

-Ayudar a que se produzca aplicaciones web más seguras.

OWASP Testing Proyect - Historia

Marzo-Abril 2003: Metodología, primera sección del documento, grupo liderado por David Endler (iDefense)

Septiembre 2003: cambio de responsable: Penny Major (OnX Enterprise Solutions Inc)

Diciembre 2003: Mark Curphey (Foundstone) toma las riendas y se decide dividir en dos fases.

Marzo 2004: comienza el trabajo sobre la Pen Testing Checklist

Junio 2004: comienza el trabajo sobre la fase dos. Daniel Cuthbert toma las riendas.

Julio 2004: Versión 1.1 de la Pen Testing Checklist.

Diciembre 2004: publicada fase 1: qué probar, técnicas y marco de referencia.

Marzo-Diciembre 2005: comienza fase 2 (objetivo: septiembre)

Marzo 2006: Nuevo responsable de proyecto: Eoin Keary.

Marzo 2006: solicitados voluntarios para fase 2.

Resumen del desarrollo

-Responsables muy ligados a proyectos de seguridad.

-Muchos cambios de responsable.

-Muchos colaboradores.

-Mucha discusión en la lista de correo.

-Algunas muy buenas ideas en la lista de correo.

-Algunos documentos producidos.

-Más documentos en desarrollo.

Algunas de las características habituales en un proyecto voluntario gestionado a través de Internet ☺

Otros proyectos relacionados

Existen otros proyectos e iniciativas con cierto solape que merece la pena mencionar. Manuales genéricos de pruebas de intrusión:

-*Open Source Security Testing Methodology Manual* (OSSTMM de ISECOM, http://www.isecom.org/osstmm/)

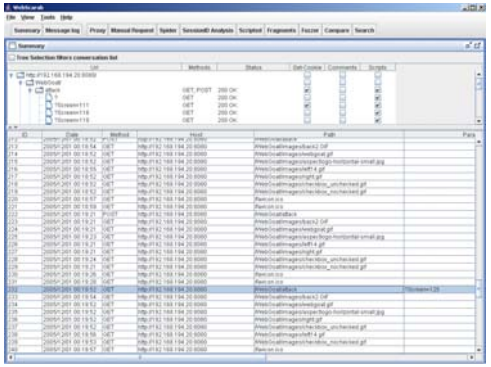
-*Information Systems Security Assessment Framework* (ISSAF de OSSIG, http://www.oissg.org/)

Cuentan con una sección dedicada a las pruebas sobre aplicaciones web. No son necesariamente competencia y en algunos casos son complementarios con los trabajos desarrollados dentro de OWASP.

Proyectos relacionados OWASP – WebScarab y WebGoat

OWASP Webscarab

Herramienta multiplataforma (Java) para realizar pruebas de aplicaciones web funcionando como proxy interceptor:



-Registra todos los accesos (documentación)

-Modificación arbitraria de peticiones y respuestas

-Extensible a través de complementos: análisis de identificadores de sesión, pruebas automáticas de parámetros (*fuzzer*), consultas SOAP

http://www.owasp.org/software/webscarab.html

Aunque no es la única herramienta de estas características (httpush, paros, spikeproxy...)

OWASP WebGoat

Aplicación J2EE basada en Tomcat desarrollada para enseñar los problemas de aplicaciones web a través de lecciones en las que se simulan vulnerabilidades en un servidor:

-Pruebas de inyección SQL, XSS

-Manipulación de campos ocultos

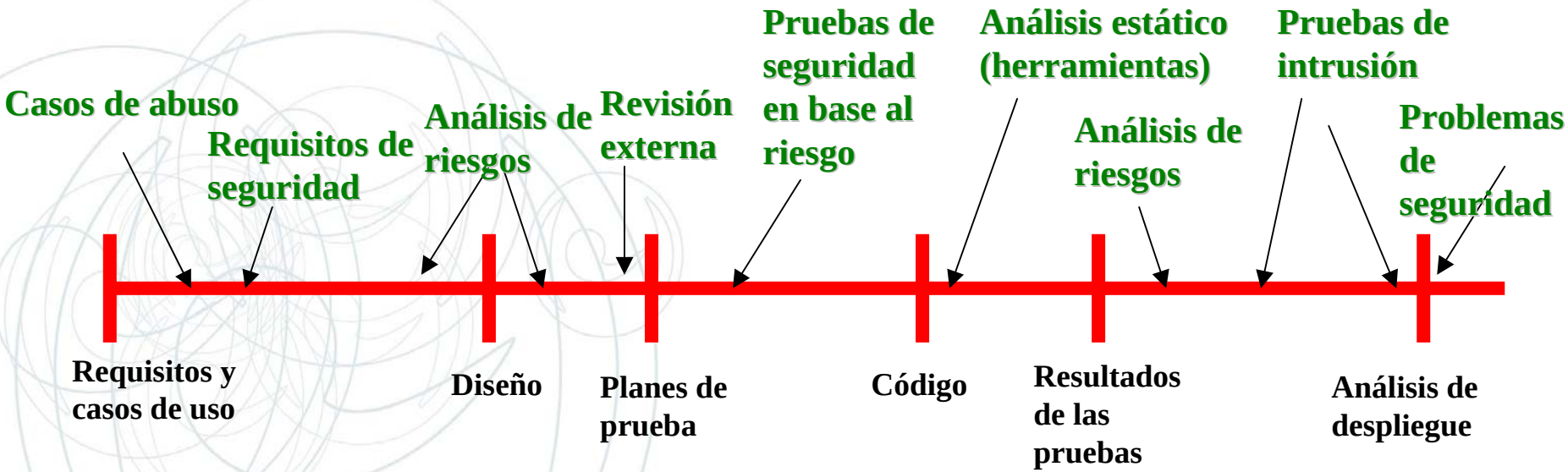
-Identificadores de sesión débiles

-....

http://www.owasp.org/software/webgoat.html

germinus

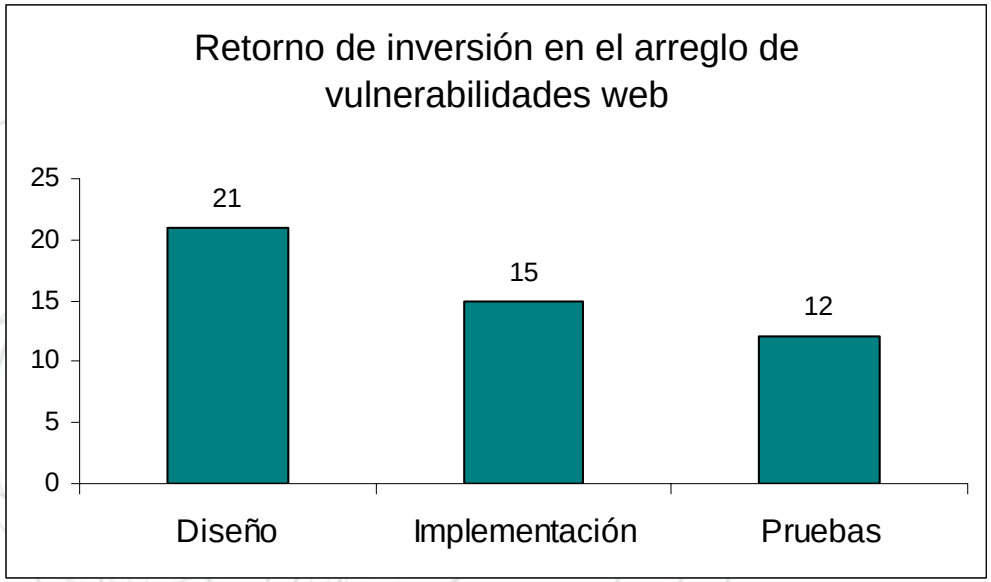
Ciclo de desarrollo seguro



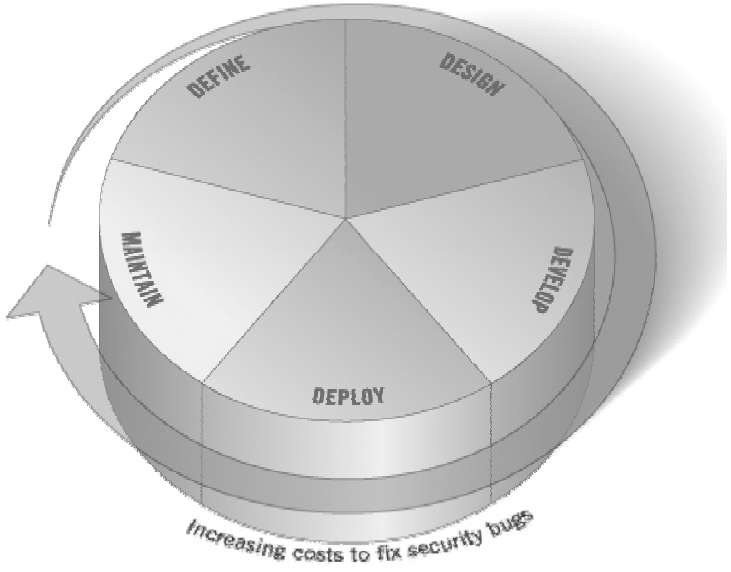
germinus

¿Cuándo arreglar vulnerabilidades?

- Es mejor identificar, buscar y arreglar los problemas de seguridad lo antes posible.
- Cuanto más tarde se arregle un problema el coste será mayor.
- Si se arreglan pronto se evitan también los costes de mantenimiento (parches).



Fuente: "Tangible ROI through software engineering" SBQ, vol 1, nº 2

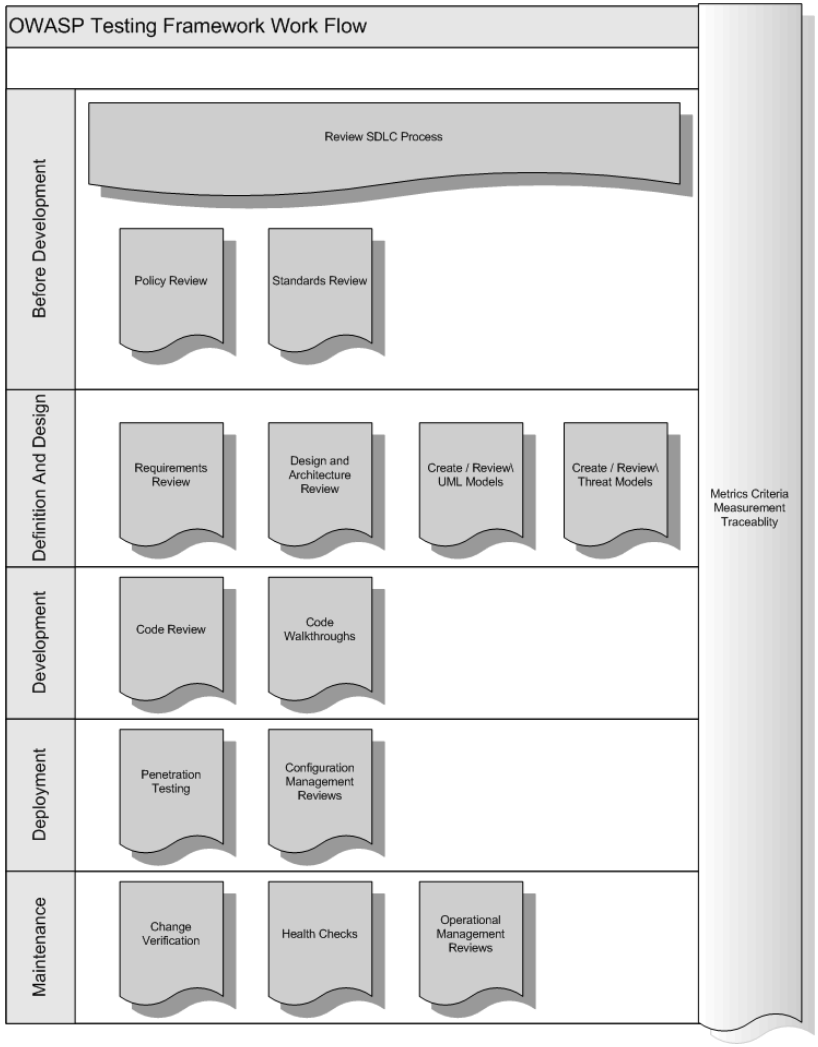


OWASP Testing Guide

Fase 1: Introducción a la metodología de pruebas de seguridad en aplicaciones web.

Fase 2: Documento que describe **cómo** realizar pruebas de seguridad en aplicaciones web.

- descripción detallada de vulnerabilidades.
- riesgos asociados.
- pruebas y mecanismos para detectarlas:
  - caja blanca.
  - caja negra.



http://www.owasp.org/index.php/Category:OWASP_Testing_Project

Ejemplo OWASP Testing Guide (fase 2)

Ficheros antiguos, de copias de seguridad o no referenciados

•Descripción del origen del problema: edición en *caliente* del sitio web, mala gestión...

•Riesgos: vulnerabilidades, exposición de información interna, código fuente...

•Cómo evitar el problema.

¿Cómo hacer las pruebas?

•**Caja negra**: inferir el nombre, fuerza bruta (al azar o metódico), obteniéndolo de otras vulnerabilidades...

•**Caja blanca**: examinar contenidos de los directorios, ficheros sin acceso, ficheros que no aparecen en los registros del servidor...

Referencias.

Ejemplos.

Herramientas.

Whitepapers.

OWASP Pen Testing Checklist

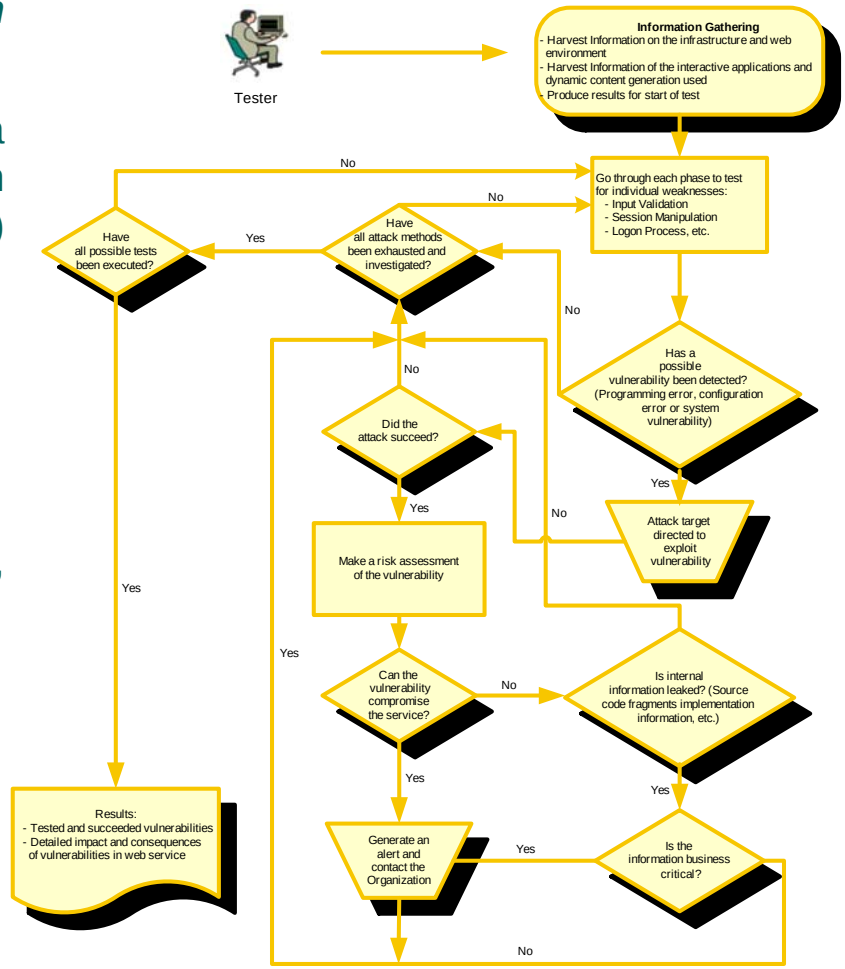
OWASP *Web Application Penetration Checklist*

Lista de pruebas a realizar para llevar a cabo una prueba de intrusión en una aplicación (reconociendo que no es siempre lo mejor) describe:

-Flujo de trabajo

-Lista de tareas a realizar y sus objetivos

Traducido a varios idiomas (Coreano, Español, Italiano)



http://www.owasp.org/documentation/testing.html

germinus

Contribuir con el proyecto OWASP:

El proyecto no sólo está abierto a contribuciones sino que las **necesita**. Contribuir es sencillo, se puede hacer:

-Enviando comentarios sobre las guías y documentación disponibles

-Probando las aplicaciones, reportando erratas y extendiéndolas

-Patrocinando un proyecto (€€€)

Trabajos futuros

Los trabajos que quieren desarrollarse a lo largo del próximo año:

-Fase 2 de la OWASP Testing Guide

Trabajos relacionados

-OWASP Legal – definir la contratación, regulación, y RFPs

¿Preguntas?

¿?

Referencias de utilidad

Algunas referencias ya comentadas:

Proyecto OWASP

http://www.owasp.org/

Proyecto OWASP en Sourceforge

http://sourceforge.net/projects/owasp

Y algunas nuevas:

Build Security In Portal

https://buildsecurityin.us-cert.gov/portal/

Secure Coding: Principles and Practices

http://www.securecoding.org/

CGI Security

http://www.cgisecurity.net/

WWW Security FAQ

http://www.w3.org/Security/Faq/

Secure Programming for UNIX and Linux HOWTO

http://www.dwheeler.com/secure-programs/

Listas de correo:

owasp-guide@sourceforge

owasp-testing@sourceforge

owasp-topten@sourceforge

owasp-dotnet@sourceforge

webappsec@securityfocus.com

pen-test@securityfocus.com

Libros recomendados

Libros recomendados:

Apache Security, Ivan Ristic, ISBN-0596007248

Core Security Patterns : Best Practices and Strategies for J2EE(TM), Web Services, and Identity Management (Core) , Christopher Steel, Ramesh Nagappan, Ray Lai, ISBN-0131463071

J2EE Security for Servlets EJBS and Web Services , Pankaj Kumar, ISBN-0131402641

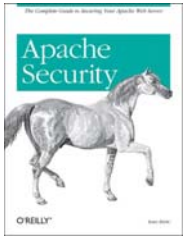
Essential PHP Security, Chris Shiflett, ISBN-059600656X

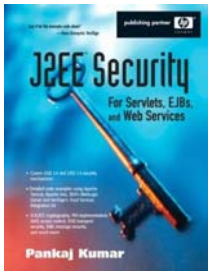
Building Secure Software: How to Avoid Security Problems the Right Way, John Viega, Gary McGraw, 020172152X

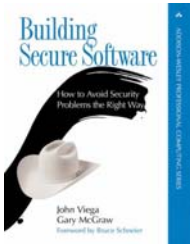
Secure Coding, Principles and Practices, Mark G. Graff, Kenneth R.Van Wyk, ISBN-0596002424

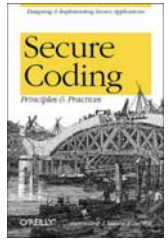
Exploiting Software, Greg Hoglund, Gary McGraw, ISBN-0201786958

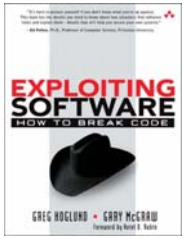
Security Engineering: A Guide to Building Dependable Distributed Systems, Ross J. Anderson, ISBN-0471389226

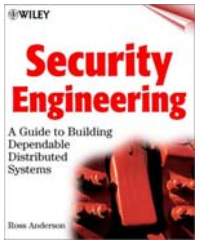












germinus

Gracias por vuestra atención