SwA Pocket Guide Status

December 2010

Software Assurance (SwA) Working Group Sessions

Tuesday, December 14, 2010

Session 4

Published Pocket Guides

- Software Assurance in Acquisition and Contract Language, Version 1.1 July 31, 2009
- Software Supply Chain Risk Management and Due Diligence, Version 1.2 June 16, 2009
- Software Assurance in Education, Training & Certification, Version 1.0, May 28, 2010 (Under revision, comments welcome)
- Software Security Testing, Version 1.0 May 10, 2010

Drafted Pocket Guides

- Architecture and Design Considerations for Secure Software, Version 0.5 October 19, 2009
- Requirements and Analysis for Secure Software, Version 1.0, October 5, 2009
- Key Practices for Mitigating the Most Egregious Exploitable Software Weaknesses, Version 1.3 May 24, 2009
- Secure Coding, Version 0.7 November 19, 2010
- SwA Business Case (outline only)

- All of these are under revision and comments are welcome

Planned Pocket Guides

- Integrating Security in the Software Development Life Cycle
- Security Considerations for Technologies, Methodologies & Languages
- Secure Software Distribution, Deployment, & Operations
- Code Transparency & Software Labels
- Assurance Case Management
- Assurance Process Improvement & Benchmarking
- Secure Software Environment & Assurance Ecosystem
- Penetration Testing throughout the Life Cycle
- Making Software Security Measurable
- Practical Measurement Framework for SwA & InfoSec

Objectives of the Secure Coding Pocket Guide (Draft)

- To describe the fundamentals of secure coding while pointing to resources for further information
  - Suggestions based on specific languages or frameworks are limited
- Stress that creating consistently secure code requires a repeatable process
- Also stress that security must be built into the other phases of the development life cycle for secure coding to be of use

Secure Coding Pocket Guide (Draft)

Key Sources and Contributors to the Secure Coding Pocket Guide

- Enhancing the Development Life Cycle to Produce Secure Software (available on the handout CD) used as starting point
- Jason Grembi, Sterling Connect, LLC, was a contributor
- Robert Seacord, CMU SEI , was a contributor
- Joe Mazzon, SRA, was a contributor and editor

Status of the Guide

- At Version 0.7 and looking for feedback
- Plan to finalize the document to Version 1.0 and have it printed

Organization of the Secure Coding Pocket Guide

- Preparing to Write Secure Code
  - Choose a Language with Security in Mind
  - Identify Safe and Secure Software Libraries
- Secure Coding Principles
  - Keep Code Small and Simple
  - Follow Secure Coding Standards and/or Guidelines
  - Use Consistent Naming
  - Use Compiler Security Checking and Enforcement
  - Review Code During and After Coding
- Secure Coding Practices
  - Validate and Encode Input
  - Filter and Sanitize Output
  - Minimize Retention of State Information
  - Do Not Allow Unauthorized Privilege Escalation
- Secure Memory and Cache Management
  - Allocate Memory and Other Resources Carefully
- Secure Error and Exception Handling
  - Integrate Anomaly Awareness
  - Use Event Monitors
- What to Avoid

Objectives of the Business Case Pocket Guide

- Provide a managerial perspective on the successful implementation of software assurance.
- Define the key components to develop a successful business case towards the integration of software assurance.
- Identify the necessities to insure propagation of value proposition.

Business Case Pocket Guide

Key Sources and Contributors

- Nancy Mead, Making the Business Case for Software Assurance
- Rafal Los, Magic Numbers: An In-Depth Guide to the 5 Key Performance Indicators for Web Application Security
- Mainstay, Fortify, Does Application Security Pay? Measuring the Business Impact of Software Security Assurance Solutions
- Viet Le, SRA, created the outline

Status of the Guide

- Working draft in progress
- A draft for review will be available Q1 2011

Organization of the Business Case Pocket Guide

- Introduction
- Risks
  - Defining Risk
  - Identifying Business Risk
  - Analyze & Prioritization
  - Validation of Risks
- Cost, Benefits, Models
  - Tangible & Intangible costs
  - Cost/Benefit models
- Metrics
  - Key Performance Indicators
  - What to measure
  - Characteristics & Type
- Process Improvement
  - Ensuring Capability
  - Levels of Maturity
- Awareness & Education

Requirements and Analysis for Secure Software (Under Revision)

Objective of this Guide

- To describe the steps and knowledge required to establish the requirements and specifications for secure software and when to apply this knowledge during the SDLC.

Proposed Changes to this Guide

- Describe the parties involved in the development of secure software requirements
- Provide a baseline set of secure software requirements

Architecture and Design Considerations for Secure Software (Under Revision)

Objective for This Guide

- To describe the steps and knowledge required to establish the architecture and high-level design for secure software during the SDLC

Proposed Changes for this Guide

- Provide architecture and design considerations for web applications (such as secure session management)
- Provide architecture and design considerations for mobile applications (such as communication security and physical security)
- Update Misuse/Abuse Case Example

Software Assurance in Education, Training & Certification

Objective for This Guide

- Current events related to cybersecurity encourage a fundamental shift in the way we think about educating and training a workforce prepared to address security issues in all phases of a software system.
- Software assurance education and training is aimed to ensure adequate coverage of requisite knowledge areas in contributing disciplines such as software engineering (including its many subdisciplines), systems engineering, project management, etc., to identify and acquire competencies associated with secure software.
- The primary audiences for this pocket guide are educators and trainers who can use this guide to identify resources to supplement their efforts as well as identify strategies to inject software assurance related topics in the existing education and training programs.

Version 1.0, May 28, 2010 (Under revision, comments welcome)

Proposed Changes for this Guide to be discussed Tuesday in Session 4A

Key Practices for Mitigating the Most Egregious Exploitable Software Weaknesses

Objective for This Guide

- This volume of pocket guide links the CWE’s with the common attacks that exploit these weaknesses, the resulting mission and business risks.
- It provides recommended practices for preventing the exploits in software.

Version 1.3 May 24, 2009

Proposed Changes for this Guide to be discussed Wednesday in Session 3B

SwA Pocket Guide Status
Discussion and Q & A

December 2010

Software Assurance (SwA) Working Group Sessions

Tuesday, December 14, 2010

Session 4