226—Secure Coding: Best Practices

North America CACS
May 8, 2012

Jim.Canup@HP.com
Abstract

Secure Coding is an action. It is an action that requires many functions of an organization to move with a common purpose. To move in this manner requires leadership and a cadence with which to move to. Without these things, there is chaos.

Secure Coding is risk driven. Different organizations have different levels of risk and therefore Secure Coding requirements should be organization specific. This delineates each organization’s definition of Secure Coding and should drive the tone and goals of their Secure Coding Strategic Plan. Without a strategic plan and a cohesive understanding of all the pieces, key components will be missed, overlooked or blatantly skipped in preference to delivery dates, functionality, profit, or just plain laziness.

Secure Coding is a journey. As with any journey, you must look around and truly understand your current location. You must also fully understand where it is that you wish to go. And only then, can you create a map capable of getting you there. Approaching it any other way will leave you wandering in the wasteland of uncertainty, listening to all who follow you shout from the darkness, “Are we there yet?”
Questions We Will Discuss

• Is Secure Code and Secure Coding the same?
• What are the components of Secure Coding?
• Who is responsible for Secure Coding?
• Where is my Secure Coding Program currently?
• How do I move my Secure Coding Program forward?
• How can I measure my progress?
Agenda

• Defining Secure Coding
• Evaluating Existing Secure Coding Efforts
• OpenSAMM
• Conducting Assessments
• Building Your Secure Coding Program
• Other “Modern” Approaches
• Quick Recap on Using SAMM
• Questions
Defining Secure Coding
Secure Code vs. Secure Coding

Secure Code = Artifact
Secure Coding = Secure SDLC
Elements of Secure Code

- Meets all technical & functional requirements
- Does not cross trust boundaries
- Properly validates all inputs
- Properly escapes all outputs
- No hardcoded personal or sensitive data
- Does not violate any of the OWASP Top 10
- Can’t be hacked!

These are Security Requirements

Jim.Canup@HP.com
Elements of Secure Coding

- Strategy & Metrics
- Policies & Compliance
- Education & Guidance

- Design Reviews
- Code Reviews
- Security Testing

- Threat Assessments
- Security Requirements
- Secure Architecture

- Vulnerability Management
- Environment Hardening
- Operational Enablement

Security Requirements are just a small part of Secure Coding
So, is Secure Code and Secure Coding the same?

**NO!**

**Secure Code** is a code artifact that meets a specified set of security requirements.

**Secure Coding** is a process used to decrease risk and increase the overall quality of code as it pertains to security.
7 Reasons Organizations Pursue a Secure Coding Program

- Organization was breached
- To meet compliance standards
- Response to an audit
- Competition - Marketplace demands it
- Customers wanting improved security
- Product/Brand improvement
- Individual crusade

Jim.Canup@HP.com
Why is this so Backwards?

<table>
<thead>
<tr>
<th>It Squeezes Profit Margins</th>
<th>It is Hard</th>
</tr>
</thead>
<tbody>
<tr>
<td>• One seventh of all companies spend less than 1% on InfoSec.</td>
<td>• Establishing a Secure Coding Program is organization-wide.</td>
</tr>
<tr>
<td>• Half of all companies spend about 6% on InfoSec.</td>
<td>• Requires special skill sets which typically necessitate outside consulting.</td>
</tr>
<tr>
<td>• The benchmark is closer to 13%.</td>
<td>• Its a continuous process – it never ends.</td>
</tr>
<tr>
<td>• Less than half ever see a concrete return on investment.</td>
<td>• It greatly minimizes risk, but it can never eliminate it.</td>
</tr>
</tbody>
</table>

Source: IT Governance: An International Guide to Data Security and Iso27001/Iso27002
Then Why Do It?

**INSURANCE** - Forrester Research, in a 2011 report, put the cost per breach between $10 and $300 million dollars.

**PRIVACY** - The 141 breaches studied resulted in an average of just under one million compromised records per breach.

**CYA** - When a breach occurs, security professionals, developers, network engineers, admins and executives fear that their job may be in jeopardy. In most cases, their fears are not far from the truth.
Review of existing secure coding efforts
CLASP

• Comprehensive, Lightweight Application Security Process
  ▫ Centered around 7 AppSec Best Practices
  ▫ Covers the entire software lifecycle (not just development)

• Adaptable to any development process
  ▫ Defines roles across the SDLC
  ▫ 24 role-based process components
  ▫ Start small and dial-in to your needs
Microsoft SDL

- Built internally for MS software
- Extended and made public for others
- MS-only versions since public release
Touchpoints

- Gary McGraw’s and Cigital’s model
Lessons Learned

• CLASP
  ▫ Large collection of activities, but no priority ordering
• Microsoft SDL
  ▫ Heavyweight, good for large ISVs
• Touchpoints
  ▫ High-level, not enough details to execute against

• ALL: Good for experts to use as a guide, but hard for non-security folks to use off the shelf
Drivers for a Maturity Model

• An organization’s behavior changes slowly over time
  ▫ Changes must be iterative while working toward long-term goals
• There is no single recipe that works for all organizations
  ▫ A solution must enable risk-based choices tailored to the organization
• Guidance related to security activities must be prescriptive
  ▫ A solution must provide enough details for non-security-people
• Overall, must be simple, well-defined, and measurable
Therefore, a viable model must...

- Define building blocks for a Secure Coding program
  - Delineate all functions within an organization that could be improved over time
- Define how building blocks should be combined
  - Make creating change in iterations a no-brainer
- Define details for each building block clearly
  - Clarify the security-relevant parts in a widely applicable way (for any org doing software development)
OpenSAMM

Software Assurance Maturity Model
http://www.opensamm.org
Understanding the model
SAMM Business Functions

• Start with the core activities tied to any organization performing software development
• Named generically, but should resonate with any developer or manager
SAMM Security Practices

- Within each of the Business Functions, 3 Security Practices are defined
- The Security Practices cover all areas relevant to secure coding
- Each one is a ‘silol' for improvement
Under each Security Practice

• Three successive Objectives & Activities under each Practice define how it can be improved over time
  ▫ This establishes a notion of a Level at which an organization fulfills a given Practice
• The three Levels for a Practice generally correspond to:
  ▫ 0: Implicit starting point with the Practice unfulfilled
  ▫ 1: Initial understanding and ad hoc provision of the Practice
  ▫ 2: Increase efficiency and/or effectiveness of the Practice
  ▫ 3: Comprehensive mastery of the Practice at scale
## Check out this one...

### Education & Guidance

<table>
<thead>
<tr>
<th>Objective</th>
<th>Activities</th>
<th>Activities</th>
<th>Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>EG 1</td>
<td>Offer development staff access to resources around the topics of secure programming and deployment</td>
<td>Educate all personnel in the software life-cycle with role-specific guidance on secure development</td>
<td>Mandate comprehensive security training and certify personnel for baseline knowledge</td>
</tr>
<tr>
<td>A. Conduct technical security awareness training</td>
<td>A. Conduct role-specific application security training</td>
<td>A. Create formal application security support portal</td>
<td>A. Establish role-based examination/certification</td>
</tr>
<tr>
<td>B. Build and maintain technical guidelines</td>
<td>B. Utilize security coaches to enhance project teams</td>
<td>B. Establish role-based examination/certification</td>
<td></td>
</tr>
</tbody>
</table>
Per Level, SAMM defines...

- Objective
- Activities
- Results
- Success Metrics
- Costs
- Personnel
- Related Levels
Approach to iterative improvement

• Since the twelve Practices are each a maturity area, the successive Objectives represent the “building blocks” for any secure coding program

• Simply put, improve a Secure Coding program in phases by:
  1. Select security Practices to improve in next phase of the program
  2. Achieve the next Objective in each Practice by performing the corresponding Activities at the specified Success Metrics
Applying the model
Conducting assessments

- SAMM includes assessment worksheets for each Security Practice

<table>
<thead>
<tr>
<th>Education &amp; Guidance</th>
<th>Yes/No</th>
</tr>
</thead>
<tbody>
<tr>
<td>✦ Have most developers been given high-level security awareness training?</td>
<td></td>
</tr>
<tr>
<td>✦ Does each project team have access to secure development best practices and guidance?</td>
<td></td>
</tr>
<tr>
<td>✦ Are most roles in the development process given role-specific training and guidance?</td>
<td>EG 1</td>
</tr>
<tr>
<td>✦ Are most stakeholders able to pull in security coaches for use on projects?</td>
<td>EG 2</td>
</tr>
<tr>
<td>✦ Is security-related guidance centrally controlled and consistently distributed throughout the organization?</td>
<td></td>
</tr>
<tr>
<td>✦ Are most people tested to ensure a baseline skill-set for secure development practices?</td>
<td>EG 3</td>
</tr>
</tbody>
</table>
Assessment process

- Supports both lightweight and detailed assessments
- Organizations may fall in between levels (+)
Creating Scorecards

- **Gap analysis**
  - Capturing scores from detailed assessments versus expected performance levels

- **Demonstrating improvement**
  - Capturing scores before and after an iteration of Secure Coding program iteration

- **Ongoing measurement**
  - Capturing scores over consistent time frames for a Secure Coding program that is already in place
Roadmap templates

- To make the “building blocks” usable, SAMM defines Roadmap templates for typical kinds of organizations
  - Independent Software Vendors
  - Online Service Providers
  - Financial Services Organizations
  - Government Organizations
- Organization types chosen because
  - They represent common use-cases
  - Each organization has variations in typical software-induced risk
  - Optimal creation of a Secure Coding program is different for each
Building Secure Coding Programs

1. Conduct initial assessment
2. Create empty roadmap?
   - Yes: Select appropriate roadmap
   - No: Existing roadmap template?
     - Yes: Adjust roadmap to organization
     - No: Adding another phase?
6. Select Practices to improve
7. Mark selected improvements on roadmap
8. Done
Case Studies

• A full walkthrough with prose explanations of decision-making as an organization improves
• Each Phase described in detail
  ▫ Organizational constraints
  ▫ Build/buy choices
• One case study exists today, several more in progress using industry partners
SAMM and the real world
SAMM history

- Beta released August 2008
  - 1.0 released March 2009
- Originally funded by Fortify Software
  - Still actively involved and using this model
- Released under a Creative Commons Attribution Share-Alike license
- Donated to OWASP and is currently an OWASP project
Expert contributions

- Built based on collected experiences with 100’s of organizations
  - Including security experts, developers, architects, development managers, IT managers

**Author & Project Lead**
Pravir Chandra

**Contributors/Reviewers**

<table>
<thead>
<tr>
<th>Fabio Arciniegas</th>
<th>Brian Chess</th>
<th>Matteo Meucci</th>
<th>John Steven</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matt Bartoldus</td>
<td>Dinis Cruz</td>
<td>Jeff Payne</td>
<td>Chad Thunberg</td>
</tr>
<tr>
<td>Sebastien Deleersnyder</td>
<td>Justin Derry</td>
<td>Gunnar Peterson</td>
<td>Colin Watson</td>
</tr>
<tr>
<td>Jonathan Carter</td>
<td>Bart De Win</td>
<td>Jeff Piper</td>
<td>Jeff Williams</td>
</tr>
<tr>
<td>Darren Challey</td>
<td>James McGovern</td>
<td>Andy Steingruebl</td>
<td></td>
</tr>
</tbody>
</table>
Industry support

• Several more case studies underway
The OpenSAMM Project

- [http://www.opensamm.org](http://www.opensamm.org)
- Dedicated to defining, improving, and testing the SAMM framework
- Always vendor-neutral, but lots of industry participation
  - Open and community driven
- Targeting new releases every 6-12 months
- Change management process
  - SAMM Enhancement Proposals (SEP)
Future plans

• Mappings to existing standards and regulations (many underway currently)
  ▫ PCI, COBIT, ISO-17799/27002, ISM3, etc.
• Additional roadmaps where need is identified
• Additional case studies
• Feedback for refinement of the model
• Translations into other languages
Other “modern” approaches

- Microsoft SDL Optimization Model
- Fortify/Cigital Building Security In Maturity Model (BSIMM)
SDL Optimization Model

- Built by MS to make SDL adoption easier
BSIMM

- Framework derived from SAMM Beta
- Based on collected data from 9 large firms

<table>
<thead>
<tr>
<th>Governance</th>
<th>Intelligence</th>
<th>SSDL Touchpoints</th>
<th>Deployment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strategy and Metrics</td>
<td>Attack Models</td>
<td>Architecture Analysis</td>
<td>Penetration Testing</td>
</tr>
<tr>
<td>Compliance and Policy</td>
<td>Security Features</td>
<td>Code Review</td>
<td>Software Environment</td>
</tr>
<tr>
<td>Training</td>
<td>Standards and</td>
<td>Security Testing</td>
<td>Configuration Management and</td>
</tr>
<tr>
<td></td>
<td>Requirements</td>
<td></td>
<td>Vulnerability Management</td>
</tr>
</tbody>
</table>
Quick Recap on Using SAMM

- Evaluate an organization’s existing software security practices
- Build a balanced secure coding program in well-defined iterations
- Demonstrate concrete improvements to a secure coding program
- Define and measure security-related activities throughout an organization
Conclusion

Secure Coding is a coordinated action involving many parts of an organization moving towards the same goal of reducing IT related risk. You must understand that Secure Coding is a journey of continuous improvement.

However, by using a tool such as OpenSAMM, you can accurately develop a Secure Coding Program that allows long-term planning and accurate measurements over time.

Without such knowledge and tools, you will be constantly asked “Are We There Yet?”
Thanks for your time! Questions?

Jim Canup, CEH
SSC – Enterprise Security - Fortify
HP-Enterprise Security Products
Bio

Jim Canup, CEH
SSC – Enterprise Security - Fortify
HP-Enterprise Security Products

A certified ethical hacker and software security professional, Jim is a recognized expert in Application Security. Jim owes his success to his life experiences and his “tell it like it is, no sugar-coating” attitude to consulting. As a result, he garners respect from security professionals, engineering teams and managers alike. Jim looks at his role as a partner and mentor who strives to teach his clients how to identify and fix problems themselves, instead of fixing things for them. His U.S. Navy background, a decade in application development and a decade of experience in AppSec allows him to relate to people at all levels of a company.

Today's security landscape is a virtual mine field, fraught with cyber terrorist, organized crime and identity thieves. A generation ago, the average person wouldn't have to worry about coming into contact with such nefarious people. Now day, all you have to do is plug in a network cable and log into your online bill pay account. The answer is education and awareness, which Jim strives to contribute to every chance he gets.