Software Assurance Maturity Model
http://www.opensamm.org

Dan Cornell
Denim Group  dan@denimgroup.com
OpenSAMM Core Team Member

Original Slides From Pravir Chandra, OpenSAMM Project Lead
chandra@owasp.org
Agenda

• Review of existing secure SDLC efforts
• Understanding the model
• Applying the model
• Exploring the model’s levels and activities
• SAMM and the real world
By the end, you’ll be able to...

• Evaluate an organization’s existing software security practices
• Build a balanced software security assurance program in well-defined iterations
• Demonstrate concrete improvements to a security assurance program
• Define and measure security-related activities throughout an organization
Review of existing secure SDLC efforts
CLASP

- Comprehensive, Lightweight Application Security Process
- Centered around 7 AppSec Best Practices
- Cover 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

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

- From each of the Business Functions, 3 Security Practices are defined
- The Security Practices cover all areas relevant to software security assurance
- Each one is a ‘silo’ for improvement
Under each Security Practice

- Three successive Objectives 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...

<table>
<thead>
<tr>
<th>Objective</th>
<th>Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>EG 1: Offer development staff access to</td>
<td>A. Conduct technical security awareness training</td>
</tr>
<tr>
<td>resources around the topics of secure</td>
<td>B. Build and maintain technical guidelines</td>
</tr>
<tr>
<td>programming and deployment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>EG 2: Educate all personnel in the software</td>
<td>A. Conduct role-specific application security training</td>
</tr>
<tr>
<td>life-cycle with role-specific guidance on</td>
<td>B. Utilize security coaches to enhance project teams</td>
</tr>
<tr>
<td>secure development</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>EG 3: Mandate comprehensive security training</td>
<td>A. Create formal application security support portal</td>
</tr>
<tr>
<td>and certify personnel for baseline knowledge</td>
<td>B. Establish role-based examination/certification</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Per Level, SAMM defines...

- Objective
- Activities
- Results
- Success Metric
- 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 assurance program

• Simply put, improve an assurance program in phases by:
  1. Select security Practices to improve in next phase of assurance 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></td>
</tr>
<tr>
<td>✦ Are most stakeholders able to pull in security coaches for use on projects?</td>
<td></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></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 from before and after an iteration of assurance program build-out
- Ongoing measurement
  - Capturing scores over consistent time frames for an assurance program that is already in place
Roadmap templates

• To make the “building blocks” usable, SAMM defines Roadmaps 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 an assurance program is different for each
Building Assurance Programs
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
Exploring the model’s levels and activities
The SAMM 1.0 release
SAMM and the real world
SAMM history

- Beta released August 2008
- 1.0 released March 2009
- Originally funded by Fortify
- 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

- Based on collected data from 9 large firms
- Recently expanded to 30

<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 and Design</td>
<td>Code Review</td>
<td>Software Environment</td>
</tr>
<tr>
<td>Training</td>
<td>Standards and Requirements</td>
<td>Security Testing</td>
<td>Configuration Management and Vulnerability Management</td>
</tr>
</tbody>
</table>
OpenSAMM Resources

- Nick Coblentz - SAMM Assessment Interview Template (xls/googledoc)
- Christian Frichot - SAMM Assessment Spreadsheet (xls)
- Colin Watson - Roadmap Chart Template (xls)
- Jim Weiler - MS Project Plan Template (mpp)
- Denim Group – Vulnerability Manager (web application)
Quick re-cap on using SAMM

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

• Use SAMM and tell us about it
• Blog, email, etc.
• Latest news at http://www.opensamm.org
• Sign up for the mailing list
Thanks for your time! Questions?

http://www.opensamm.org

Dan Cornell
Denim Group dan@denimgroup.com
OpenSAMM Core Team Member

Original Slides From Pravir Chandra, OpenSAMM Project
chandra@owasp.org