

OWASP Software Assurance Maturity Model (SAMM) Version 2.0 Update

John DiLeo, OWASP SAMM Project Team February 2019

About Me

- Born and raised in northeastern US
- Spent A LOT of time in school
- First career: Operations Research / Simulation
- Second career: Web Development
- Third Career: Application Security, since 2014
 - Focus on Software Assurance
 - Moved to NZ in 2017, joined Orion Health
 - Active in OWASP in US and NZ
- Joined OWASP SAMM team in June



What is SAMM?

- The Software Assurance Maturity Model (SAMM) is an open framework to help organizations formulate and implement a strategy for software security that is tailored to the specific risks facing the organization.
- The resources provided by SAMM will aid in:
 - Evaluating an organization's existing software security practices.
 - Building a balanced software security assurance program in well-defined iterations.
 - Demonstrating concrete improvements to a security assurance program.
 - Defining and measuring security-related activities throughout an organization.



Why SAMM?

"The most that can be expected from any model is that it can supply a useful approximation to reality: All models are wrong; some models are useful." – George E. P. Box



Core Principles of SAMM

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

Guidance related to security activities must be prescriptive

Overall, must be simple, well-defined, and measurable

 A solution must enable risk-based choices tailored to the organization

 A solution must provide enough details for non-security-people

 OWASP Software Assurance Maturity Model (SAMM)



☆ 🔍 🕕 : v1.5 v2.0 Beta

Introduction

The mission of OWASP SAMM is to be the prime maturity model for software assurance that provides an effective and measurable way for all types of organizations to analyse and improve their software security posture. OWASP SAMM supports the complete software lifecycle, including development and acquisition, and is technology and process agnostic. It is intentionally built to be evolutive and risk-driven in nature.

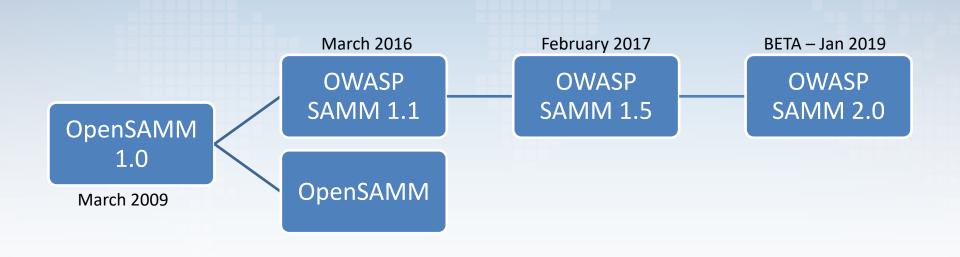
The original model (v1.0) was written by Pravir Chandra and dates back from 2009. Over the last 10 years, it has proven a widely distributed and effective model for improving secure software practices in different types of organisations throughout the world. Translations and supporting tools have been contributed by the community to facilitate adoption and alignment. With version 2.0, we further improve the model to deal with some of its current limitations.

After a period of intensive discussions and with input from practitioners and the OWASP community during summits in Europe and the US on the best way forward, we take a new approach for version 2.0 based on the following improvements.

• The model is development paradigm agnostic, it supports waterfall, iterative, agile and



Project History





The Core Team

- Sebastien (Seba) Deleersnyder Project Leader, Belgium
- Chris Cooper Webmaster, United Kingdom
- Bart DeWin Belgium
- John DiLeo New Zealand
- Daniel Kefer Germany
- Nessim Kisserli United Kingdom
- Yan Kravchenko United States



The Core Framework

Version 1.5

Four Business Functions

- Governance
- Construction
- Verification
- Operations

Version 2.0

Adds a Fifth Business Function

- Governance
- Design
- Implementation
- Verification
- Operations



The Security Practices

Governance

- Strategy & Metrics
- Policy & Compliance
- Education & Guidance

Design

- Threat Assessment
- Security Requirements
- Security Architecture

Implementation

- Secure Build
- Secure Deployment
- Defect Management

Verification

- Architecture Assessment
- Requirements-Driven Testing
- Security Testing

• Operations

- Incident Management
- Environment Management
- Operational Management



The Maturity Levels

OWASP SAMM - 3 levels

- Level 1
- Level 2
- Level 3

Rough alignment with CMMI levels Initial

- 2(a) (Partially) Managed
- 2(b) (Fully) Managed
- 3 Defined

1

- 4 Quantitatively Managed
- 5. Optimising



Activity Streams Example – Operational Management

A: Data Protection

Level 1: Basic Data Protections in Place Level 2: Data cataloged and data protection policy established Level 3: Data policy breaches detected and acted upon B: System Decomm / Legacy Management Level 1: Identification of unused apps/services Level 2: Decommissioning

- and legacy migration processes in place
- Level 3: Proactive handling of legacy applications/ services



Pain Points with Scoring in SAMM 1.5

Strategy & Metrics, Level 1: *Is there a software security assurance program in place?*

Available Responses:

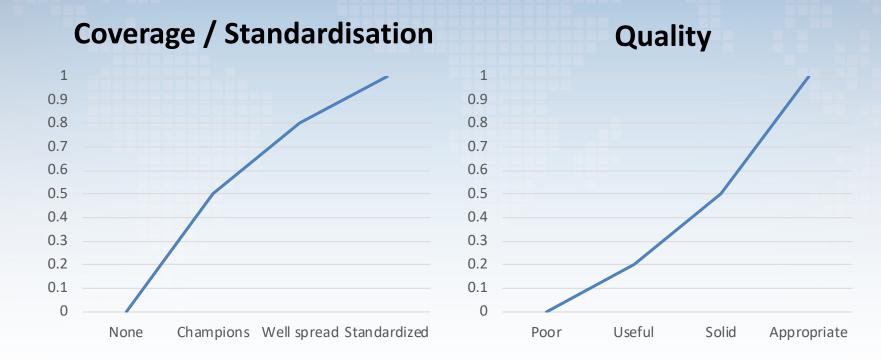
- No
- Yes, it's less than a year old
- Yes, it's a number of years old
- Yes, it's a pretty aature program

But, what about...

- Quality of the Programme?
- Currency of the Programme? Has it been reviewed/updated?
- How do you know the program is still relevant?

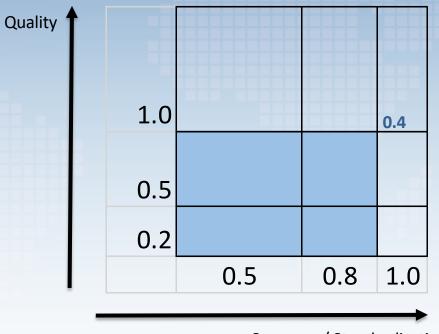


Consider Multiple Dimensions





Combining Dimension Scores

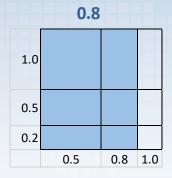


Coverage / Standardization

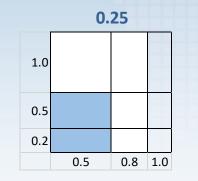
MATURITY SCORE = QUALITY × COVERAGE



Education & Guidance Practise



Level 1 Relevant employees are provided an awareness training



Level 2 Employees are provided role specific trainings

0.0

Level 3 Employee's knowledge is regularly assessed



Open Questions

- Number of response values for quality and coverage questions
 - Four? Five?
 - Linear?
- How to compute overall maturity score from individual metric scores across levels

- Level 2 way more expensive than Level 1



Interested in Getting Involved?

- Provide comments on the current draft
 - <u>https://owaspsamm.org/v2.0b/feedback/</u>
- Join our monthly project calls
 - Second Wednesday of the month, 9:30 p.m. Central European Time
 - That translates to Thursday morning, at 7:30 or 9:30 a.m.
- Join our Slack Channel
 - #project-samm on the OWASP Slack (<u>https://owasp.slack.com/</u>)

