NIST Consumer Software Labeling

Leveraging the OWASP Software Assurance Maturity Model (SAMM)

Submitted by Brian Glas, Assistant Professor of Computer Science,
Co-lead of OWASP Top 10 and lead for OWASP SAMM Benchmark
https://www.linkedin.com/in/brianglas/
brian.glas@gmail.com

OWASP SAMM (https://owaspsamm.org) can directly contribute to the *formal and informal processes and practices used to secure the software development process*. SAMM provides a practical and measurable way for organizations to analyze and improve their software security posture. We want to raise awareness and educate organizations on designing, developing, and deploying secure software through our model. SAMM supports the complete software lifecycle and is technology and process agnostic. We built SAMM to be evolutive and risk-driven, as no single recipe works for all organizations.

SAMM is a comprehensive model built on five core business functions: Governance, Design, Implementation, Verification, and Operations. The scope of SAMM is more than just a Software Development Lifecycle (SDL); it includes Governance to help with the program and process supporting the SDL, and Operations which is either left out of an SDL or is a footnote. SAMM is a maturity model where business functions contain security practices, and within those practices are activities grouped in streams that build on each other as the organization matures.

SAMM's scoring model is based on 0-1 for each activity: 0.0-none, 0.2-few, 0.5-half, 1.0-most. We built the model to show incremental improvements in their activities, understanding they cannot go from 0-No to 1-Yes in a single step. We believe an organization needs to show incremental improvements by metrics to help justify the investment in initiatives and projects, and the model keeps this mantra in mind. This scoring model should satisfy the *technical criteria needed to support validation of consumer software security assertions that reflect a baseline level of secure practices*.

We used SAMM as the core for secure development process assessments in the recent RABET-V pilot project (https://github.com/it-dept-cis/RABET-V-Pilot). RABET-V is a flexible, risk-based, and cost-effective election system verification process that will expedite election system verification while providing security, reliability, and usability assurances. The RABET-V Pilot Program is designed to evaluate the RABET-V process and the potential to improve the speed, security assurances, and cost-effectiveness of non-voting election technology verification. We augmented SAMM with practices and activities for usability and accessibility, and so far, it's been very promising in the results.

We are working on the SAMM Benchmark Project to collect SAMM maturity scores from assessments and build a population to enable comparisons based on size, geographic area, industry, and other metadata. This dataset will provide a valuable resource for organizations to better understand their security maturity level compared to peers. Historically, this visibility and comparison can help drive improvements across industries as organizations "compete" to earn a top score. As long as the model contains desired practices and activities conducive to improving security and managing risk, the competition to obtain higher scores benefits organizations, industries, and ultimately consumers. For the goal of *measures for incentivizing participation by consumer software developers*, if the labels are designed in a manner that is viewed as a competitive advantage that positively influences consumers to purchase that product and drives the desired behavior of adoption of security practices and activities, then it should produce valuable results.

I firmly believe that SAMM can provide a foundation for the efforts related to building, measuring, and communicating the security maturity of products. The SAMM model supports both self-assessment and third-party assessments today and is widely used in both ways. While SAMM Assessments are not typically conducted as audits, organizations can provide evidence that answers represent the current state of practices and activities related to software assurance. Different styles of SAMM Assessments can help support *how different conformity assessment approaches (e.g., vendor attestation, third-party conformity assessment) can be employed in consumer software labeling efforts*. The SAMM team is already planning to differentiate self-assessment and third-party assessment in the SAMM Benchmark. We believe there is a higher level of assurance in the third-party process and scoring. SAMM is designed in a balanced, modular fashion that can be augmented as needed.

For *consumer product labeling programs for educating the public on the security properties of consumer software*, this will be a challenge due to a lack of shared understanding of what security properties are and their benefits. I expect it to require a focused awareness program to help consumers understand the different security properties to make informed decisions and the labeling has the desired effect or benefit.

Lastly, addressing *feasibility and possible means for implementing tiered labels that reflect increasingly comprehensive levels of testing and assessment.* I believe this is only useful as a subset of the full software assurance model, as we should have learned by now that we cannot test ourselves secure. Only a complete software assurance program can hope to manage the software risk effectively and efficiently.