OWASP
Open Web Application
Security Project

OWASP
SAMM

OWASP SAMM Update

SAMM User Day
June 16th, 2020
Bart De Win, Seba Deleersnyder

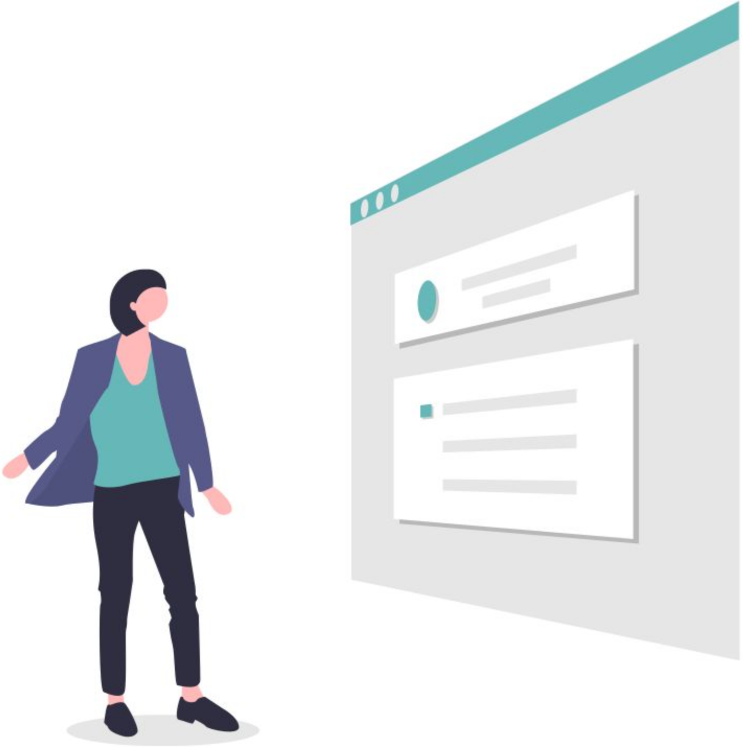
What is SAMM?

OWASP
FLAGSHIP
mature projects

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 analyze 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.

Visit our website

owaspsamm.org

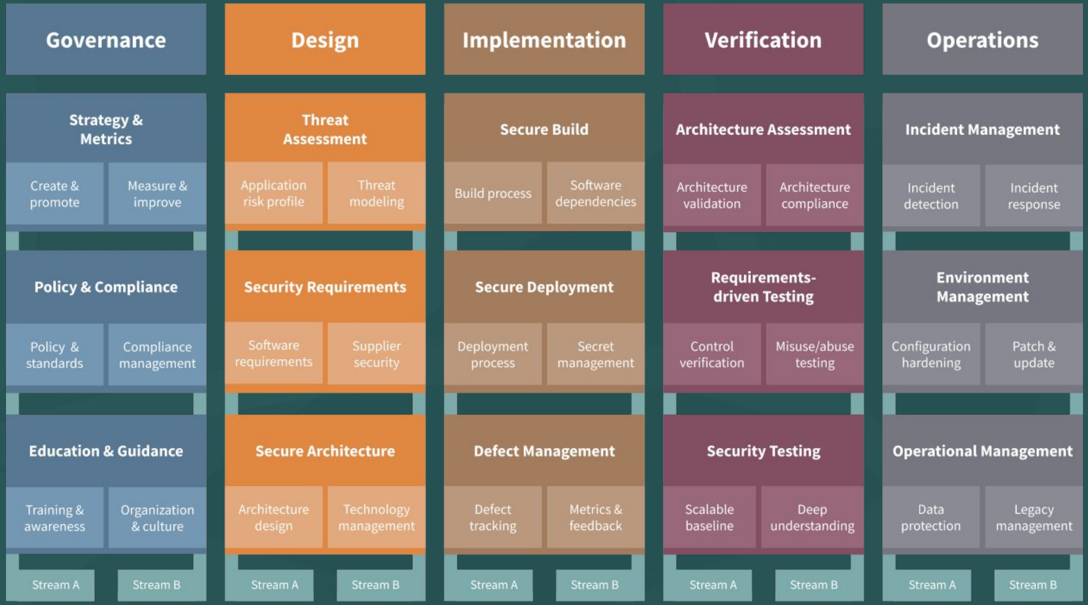


Goals of SAMM version 2

- Align with recent development practices
- Revise all activities (no “orphans”)
- Method agnostic
- Improve assessments
- Improve production process

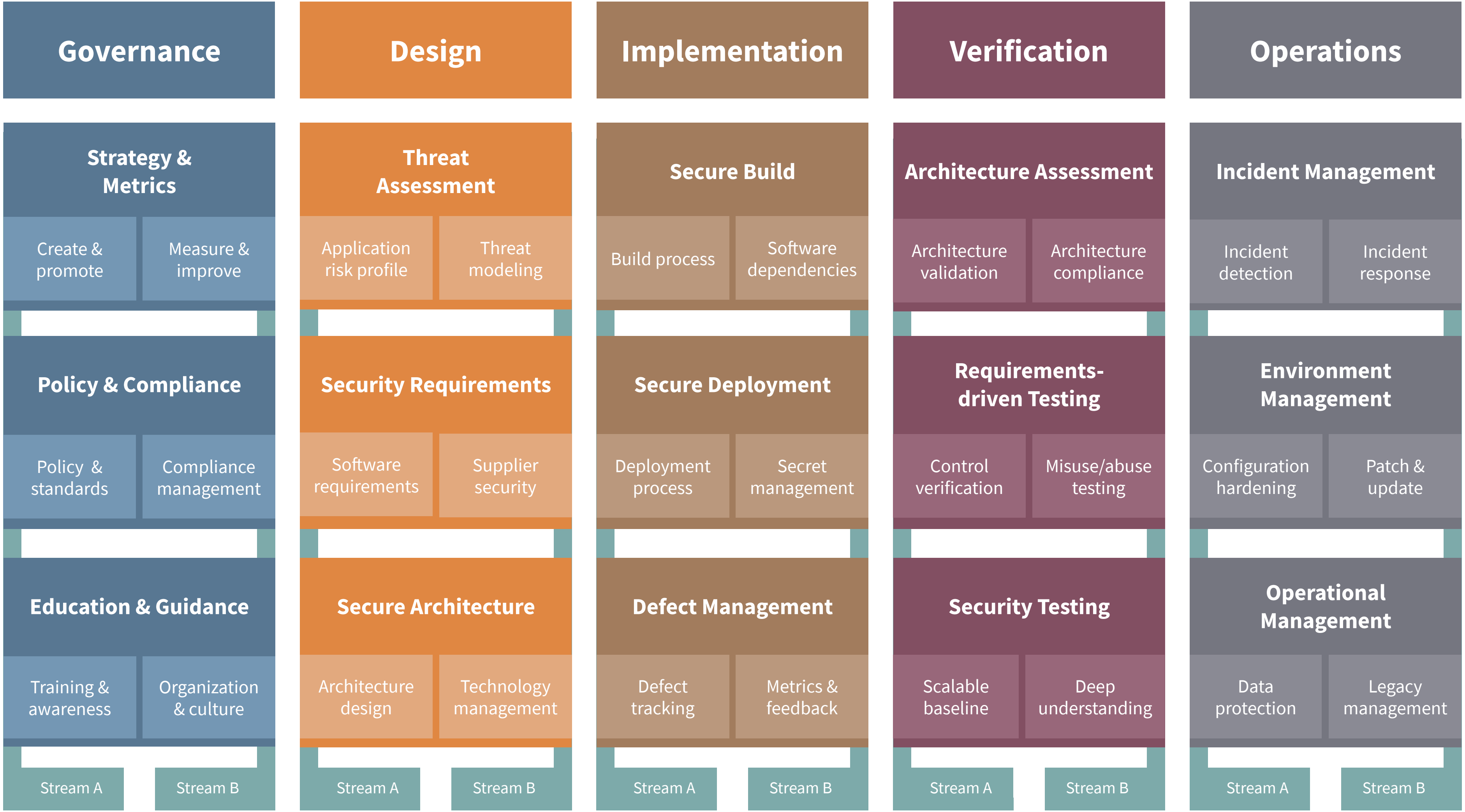
Backwards compatibility was not a goal

Core structure



Governance		Design		Implementation		Verification		Operations	
Strategy & Metrics		Threat Assessment		Secure Build		Architecture Assessment		Incident Management	
Create & promote	Measure & improve	Application risk profile	Threat modeling	Build process	Software dependencies	Architecture validation	Architecture compliance	Incident detection	Incident response
Policy & Compliance		Security Requirements		Secure Deployment		Requirements-driven Testing		Environment Management	
Policy & standards	Compliance management	Software requirements	Supplier security	Deployment process	Secret management	Control verification	Misuse/abuse testing	Configuration hardening	Patch & update
Education & Guidance		Secure Architecture		Defect Management		Security Testing		Operational Management	
Training & awareness	Organization & culture	Architecture design	Technology management	Defect tracking	Metrics & feedback	Scalable baseline	Deep understanding	Data protection	Legacy management
Stream A	Stream B	Stream A	Stream B	Stream A	Stream B	Stream A	Stream B	Stream A	Stream B

OWASP
Open Web Application
Security Project



	Governance	Design	Implementation	Verification	Operations
	Strategy & Metrics	Threat Assessment	Secure Build	Architecture Assessment	Incident Management
Stream A	Create & promote	Application risk profile	Build process	Architecture validation	Incident detection
Stream B	Measure & improve	Threat modeling	Software dependencies	Architecture compliance	Incident response
	Policy & Compliance	Security Requirements	Secure Deployment	Requirements-driven Testing	Environment Management
Stream A	Policy & standards	Software requirements	Deployment process	Control verification	Configuration hardening
Stream B	Compliance management	Supplier security	Secret management	Misuse/abuse testing	Patch & update
	Education & Guidance	Secure Architecture	Defect Management	Security Testing	Operational Management
Stream A	Training & awareness	Architecture design	Defect tracking	Scalable baseline	Data protection
Stream B	Organization & culture	Technology management	Metrics & feedback	Deep understanding	Legacy management

SAMM v2
security practice structure

	A: Control Verification	**B: Misuse /Abuse Testing**
Level 1 - ***Opportunistically*** *find* ***basic*** *vulnerabilities and other security issues.*	Test for standard security controls	Perform security fuzzing testing
Level 2 - *Perform* ***implementation review*** *to discover* ***application-specific risks*** *against the security requirements.*	Derive test cases from known security requirements	Create and test abuse cases and business logic flaw test
Level 3 - ***Maintain*** *the application security level after bug fixes, changes or during maintenance*	Perform regression testing (with security unit tests)	Denial of service and security stress testing

SAMM v2
security practice structure

MATURITY

	A: Control Verification	B: Misuse /Abuse sting
Level 1 *-* ***Opportunistically*** *find* ***basic*** *vulnerabilities and other security issues.*	Test for standard se controls	form security fuzzing testing
Level 2 *- Perform* ***implementation review*** *to discover* ***application-specific risks*** *against the security requirements.*	Derive test cases fr security requi	te and test abuse cases ness logic flaw test
Level 3 *-* ***Maintain*** *the application security level after bug fixes, changes or during maintenance*	Perform regression test (with security unit tests)	Denial of service and security stress testing

SAMM v2
security practice structure

STREAMS

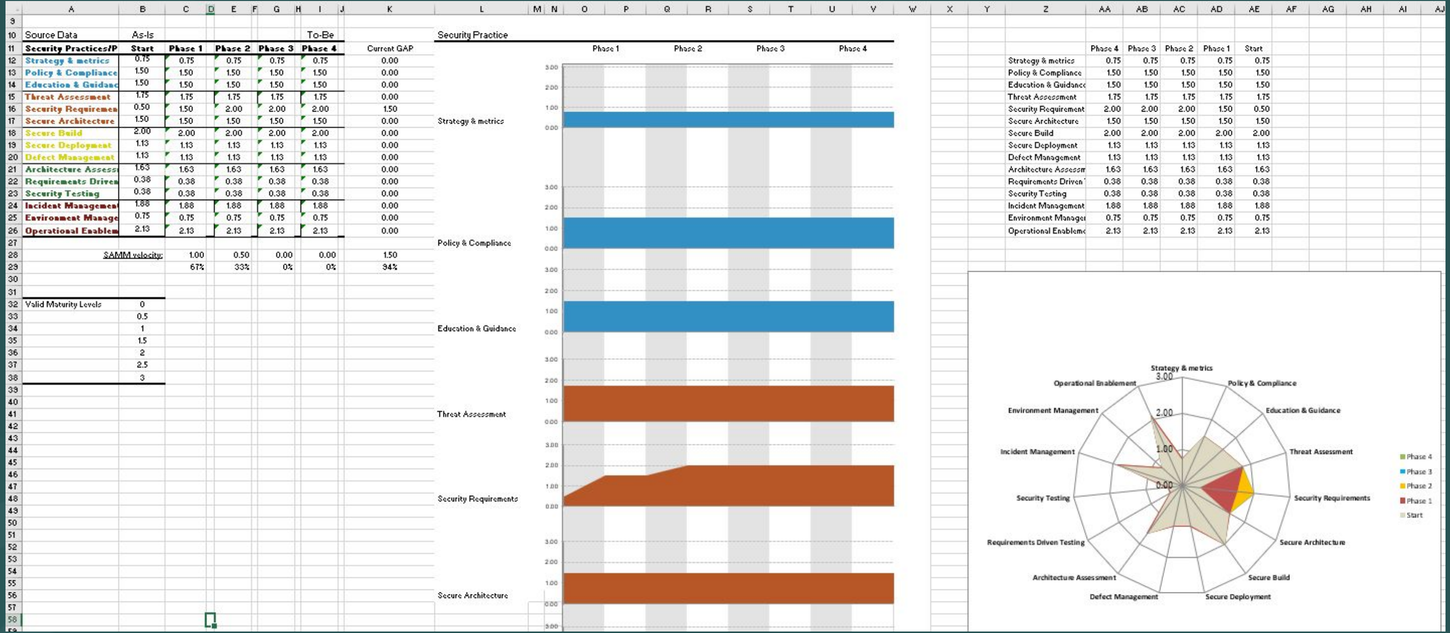
	A: Control Verification	B: Misuse /Abuse Testing
Level 1 - ***Opportunistically*** *find* ***basic*** *vulnerabilities and other security issues.*	Test for sta[illegible]ecurity controls	Perform sec[illegible]zzing testing
Leve[illegible] [illegible] ***n review*** *to discover* ***application-specific risks*** *against the security requirements.*	Derive test[illegible]om known security re[illegible]nts	Create and t[illegible]se cases and busines[illegible]law test
Level 3 - ***Maintain*** *the application security level after bug fixes, changes or during maintenance*	Perform re[illegible] testing (with security [illegible]nit tests)	Denial of se[illegible]d security stress testing

SAMM v2 assessment toolbox

GOVERNANCE		
Stream	**Level**	**Strategy and metrics**
Create and promote	1	Has the organization defined a set of risks to prioritize applications by?
		● You have captured the risk appetite of your organization’s executive leadership ● The organization’s leadership have vetted and approved risks ● You have identified the main business and technical threats to your organization’s assets and data ● Risks are documented and accessible to relevant stakeholders

https://github.com/OWASP/samm/tree/master/Supporting%20Resources/v2.0/toolbox

SAMM roadmaps



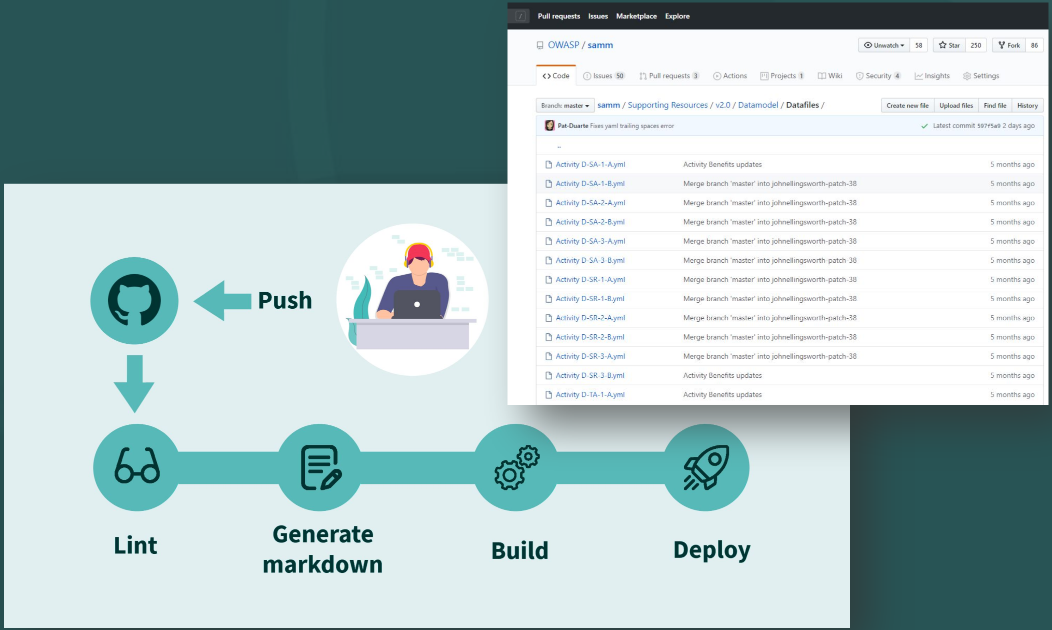
Source Data	As-Is				To-Be	
Security Practices/P	**Start**	**Phase 1**	**Phase 2**	**Phase 3**	**Phase 4**	Current GAP
Strategy & metrics	0.75	0.75	0.75	0.75	0.75	0.00
Policy & Compliance	1.50	1.50	1.50	1.50	1.50	0.00
Education & Guidanc	1.50	1.50	1.50	1.50	1.50	0.00
Threat Assessment	1.75	1.75	1.75	1.75	1.75	0.00
Security Requiremen	0.50	1.50	2.00	2.00	2.00	1.50
Secure Architecture	1.50	1.50	1.50	1.50	1.50	0.00
Secure Build	2.00	2.00	2.00	2.00	2.00	0.00
Secure Deployment	1.13	1.13	1.13	1.13	1.13	0.00
Defect Management	1.13	1.13	1.13	1.13	1.13	0.00
Architecture Assess	1.63	1.63	1.63	1.63	1.63	0.00
Requirements Driven	0.38	0.38	0.38	0.38	0.38	0.00
Security Testing	0.38	0.38	0.38	0.38	0.38	0.00
Incident Managemen	1.88	1.88	1.88	1.88	1.88	0.00
Environment Manage	0.75	0.75	0.75	0.75	0.75	0.00
Operational Enablem	2.13	2.13	2.13	2.13	2.13	0.00
	SAMM velocity:	1.00	0.50	0.00	0.00	1.50
		67%	33%	0%	0%	94%

Valid Maturity Levels	0
	0.5
	1
	1.5
	2
	2.5
	3

	Phase 4	Phase 3	Phase 2	Phase 1	Start
Strategy & metrics	0.75	0.75	0.75	0.75	0.75
Policy & Compliance	1.50	1.50	1.50	1.50	1.50
Education & Guidanc	1.50	1.50	1.50	1.50	1.50
Threat Assessment	1.75	1.75	1.75	1.75	1.75
Security Requirement	2.00	2.00	2.00	1.50	0.50
Secure Architecture	1.50	1.50	1.50	1.50	1.50
Secure Build	2.00	2.00	2.00	2.00	2.00
Secure Deployment	1.13	1.13	1.13	1.13	1.13
Defect Management	1.13	1.13	1.13	1.13	1.13
Architecture Assessm	1.63	1.63	1.63	1.63	1.63
Requirements Driven	0.38	0.38	0.38	0.38	0.38
Security Testing	0.38	0.38	0.38	0.38	0.38
Incident Management	1.88	1.88	1.88	1.88	1.88
Environment Manage	0.75	0.75	0.75	0.75	0.75
Operational Enableme	2.13	2.13	2.13	2.13	2.13

Project: SAMM CI/CD

- Single source of the truth (Github)
- Used to generate everything *automatically*
  - Document, website
  - Toolbox
  - Applications



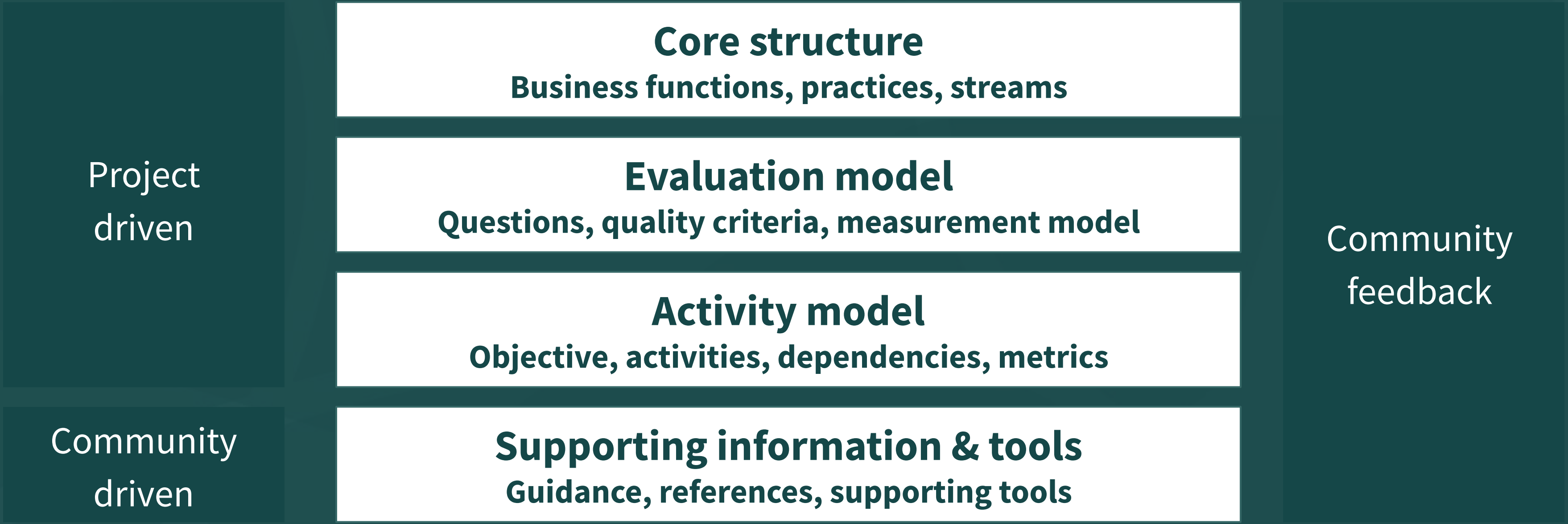
samm / Supporting Resources / v2.0 / Datamodel / Datafiles /

Pat-Duarte Fixes yaml trailing spaces error — Latest commit 597f5a9 2 days ago

..

Activity D-SA-1-A.yml	Activity Benefits updates	5 months ago
Activity D-SA-1-B.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SA-2-A.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SA-2-B.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SA-3-A.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SA-3-B.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SR-1-A.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SR-1-B.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SR-2-A.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SR-2-B.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SR-3-A.yml	Merge branch 'master' into johnellingsworth-patch-38	5 months ago
Activity D-SR-3-B.yml	Activity Benefits updates	5 months ago
Activity D-TA-1-A.yml	Activity Benefits updates	5 months ago

Community involvement



Project driven

Community driven

Core structure
Business functions, practices, streams

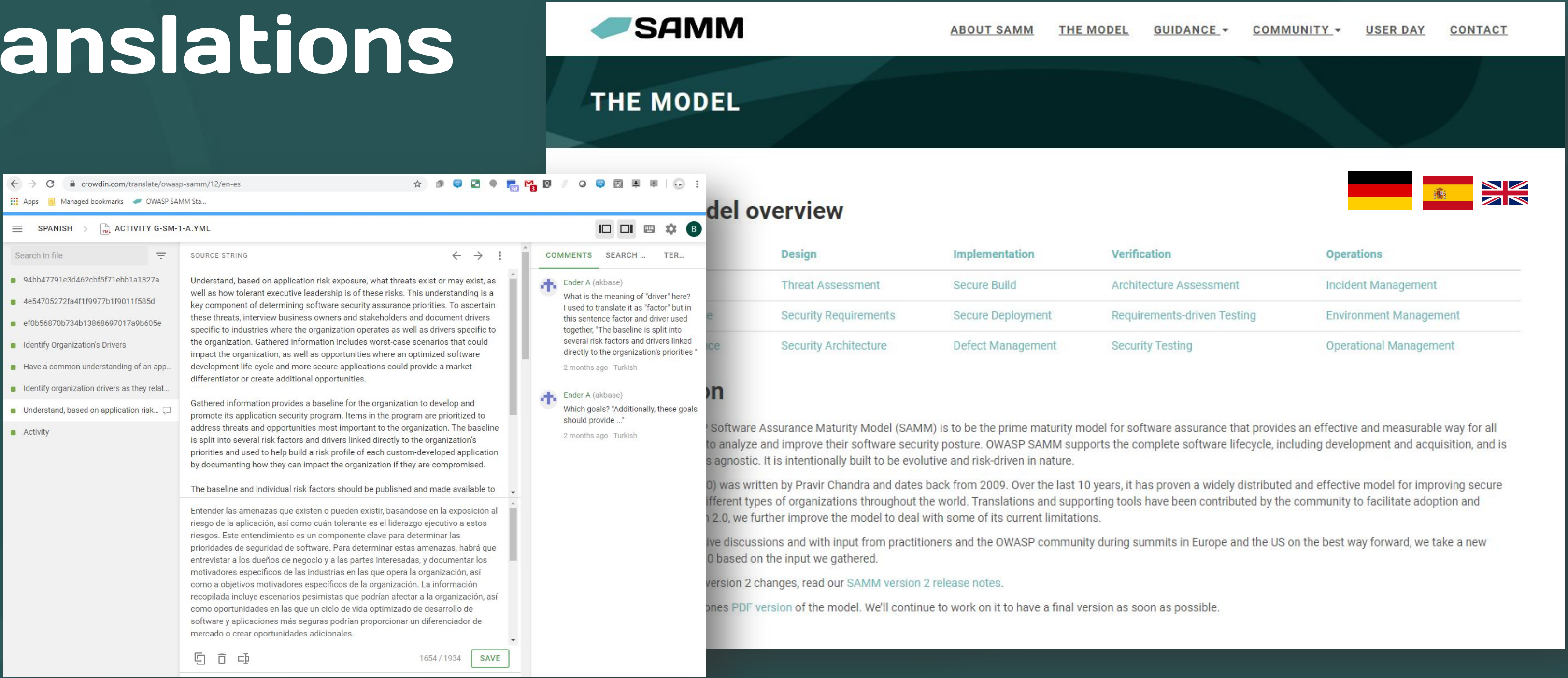
Evaluation model
Questions, quality criteria, measurement model

Activity model
Objective, activities, dependencies, metrics

Supporting information & tools
Guidance, references, supporting tools

Community feedback

Translations



- 94bb47791e3d462cbf5f71ebb1a1327a
- 4e54705272fa4f1f9977b1f9011f585d
- ef0b56870b734b13868697017a9b605e
- Identify Organization's Drivers
- Have a common understanding of an app...
- Identify organization drivers as they relat...
- Understand, based on application risk...
- Activity

Understand, based on application risk exposure, what threats exist or may exist, as well as how tolerant executive leadership is of these risks. This understanding is a key component of determining software security assurance priorities. To ascertain these threats, interview business owners and stakeholders and document drivers specific to industries where the organization operates as well as drivers specific to the organization. Gathered information includes worst-case scenarios that could impact the organization, as well as opportunities where an optimized software development life-cycle and more secure applications could provide a market-differentiator or create additional opportunities.

Gathered information provides a baseline for the organization to develop and promote its application security program. Items in the program are prioritized to address threats and opportunities most important to the organization. The baseline is split into several risk factors and drivers linked directly to the organization's priorities and used to help build a risk profile of each custom-developed application by documenting how they can impact the organization if they are compromised.

The baseline and individual risk factors should be published and made available to

Entender las amenazas que existen o pueden existir, basándose en la exposición al riesgo de la aplicación, así como cuán tolerante es el liderazgo ejecutivo a estos riesgos. Este entendimiento es un componente clave para determinar las prioridades de seguridad de software. Para determinar estas amenazas, habrá que entrevistar a los dueños de negocio y a las partes interesadas, y documentar los motivadores específicos de las industrias en las que opera la organización, así como a objetivos motivadores específicos de la organización. La información recopilada incluye escenarios pesimistas que podrían afectar a la organización, así como oportunidades en las que un ciclo de vida optimizado de desarrollo de software y aplicaciones más seguras podrían proporcionar un diferenciador de mercado o crear oportunidades adicionales.

Ender A (akbase)
What is the meaning of "driver" here? I used to translate it as "factor" but in this sentence factor and driver used together, "The baseline is split into several risk factors and drivers linked directly to the organization's priorities "
2 months ago Turkish

Ender A (akbase)
Which goals? "Additionally, these goals should provide ..."
2 months ago Turkish

THE MODEL

del overview

	Design	Implementation	Verification	Operations
	Threat Assessment	Secure Build	Architecture Assessment	Incident Management
e	Security Requirements	Secure Deployment	Requirements-driven Testing	Environment Management
ce	Security Architecture	Defect Management	Security Testing	Operational Management

on

Software Assurance Maturity Model (SAMM) is to be the prime maturity model for software assurance that provides an effective and measurable way for all to analyze and improve their software security posture. OWASP SAMM supports the complete software lifecycle, including development and acquisition, and is s agnostic. It is intentionally built to be evolutive and risk-driven in nature.

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 ifferent types of organizations throughout the world. Translations and supporting tools have been contributed by the community to facilitate adoption and 2.0, we further improve the model to deal with some of its current limitations.

ive 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 0 based on the input we gathered.

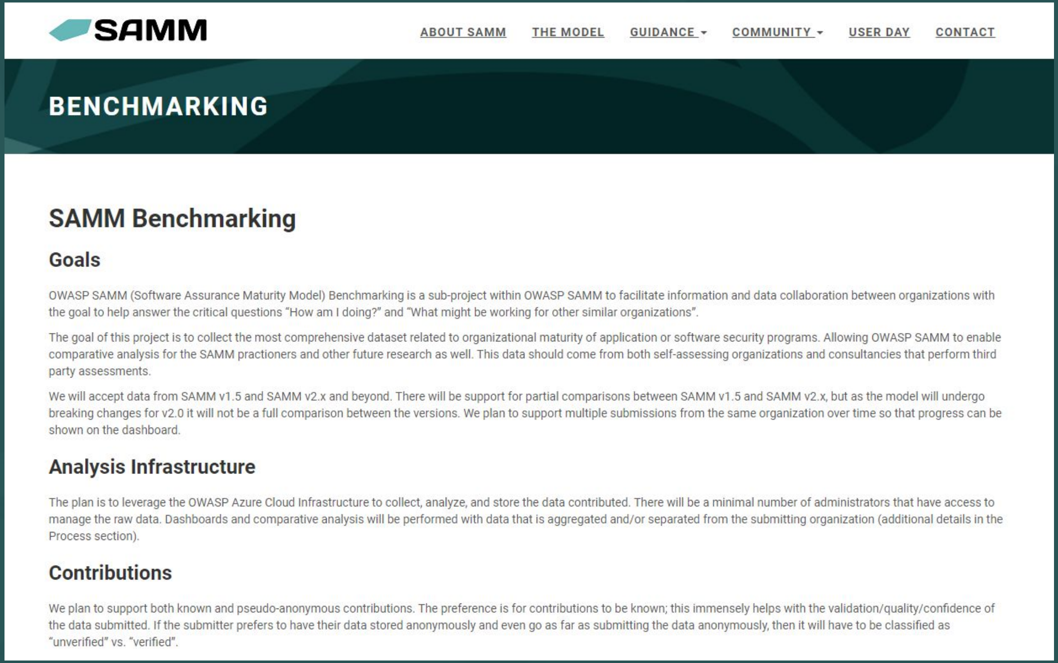
ersion 2 changes, read our SAMM version 2 release notes.

ones PDF version of the model. We'll continue to work on it to have a final version as soon as possible.

https://crowdin.com/project/owasp-samm

OWASP
Open Web Application Security Project

How do I compare?



SAMM Benchmarking

Goals

OWASP SAMM (Software Assurance Maturity Model) Benchmarking is a sub-project within OWASP SAMM to facilitate information and data collaboration between organizations with the goal to help answer the critical questions "How am I doing?" and "What might be working for other similar organizations".

The goal of this project is to collect the most comprehensive dataset related to organizational maturity of application or software security programs. Allowing OWASP SAMM to enable comparative analysis for the SAMM practioners and other future research as well. This data should come from both self-assessing organizations and consultancies that perform third party assessments.

We will accept data from SAMM v1.5 and SAMM v2.x and beyond. There will be support for partial comparisons between SAMM v1.5 and SAMM v2.x, but as the model will undergo breaking changes for v2.0 it will not be a full comparison between the versions. We plan to support multiple submissions from the same organization over time so that progress can be shown on the dashboard.

Analysis Infrastructure

The plan is to leverage the OWASP Azure Cloud Infrastructure to collect, analyze, and store the data contributed. There will be a minimal number of administrators that have access to manage the raw data. Dashboards and comparative analysis will be performed with data that is aggregated and/or separated from the submitting organization (additional details in the Process section).

Contributions

We plan to support both known and pseudo-anonymous contributions. The preference is for contributions to be known; this immensely helps with the validation/quality/confidence of the data submitted. If the submitter prefers to have their data stored anonymously and even go as far as submitting the data anonymously, then it will have to be classified as "unverified" vs. "verified".

Our roadmap

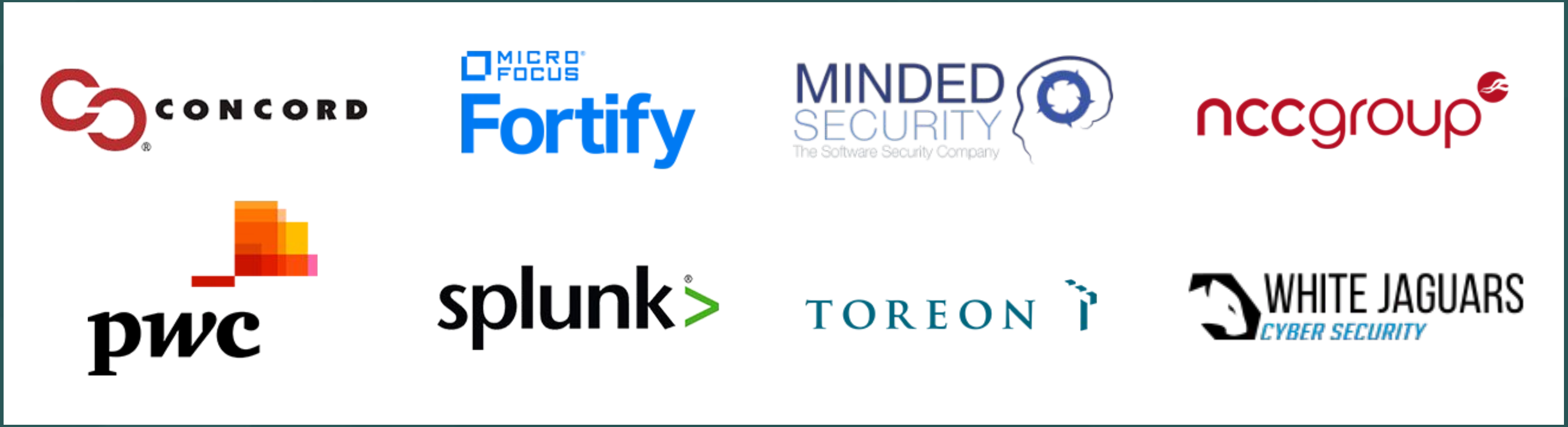
- Continuous: minor fixes
- Wrap-up: PDF
- v2.1 (Oct 2020): Translations, mappings
- v2.2 (Jan 2021): Activity-specific guidance (references, agile, ...)
- V2.3 (June 2021): online toolbox, open API
- V3.0: tbd

Let’s do this together

Who is SAMM?

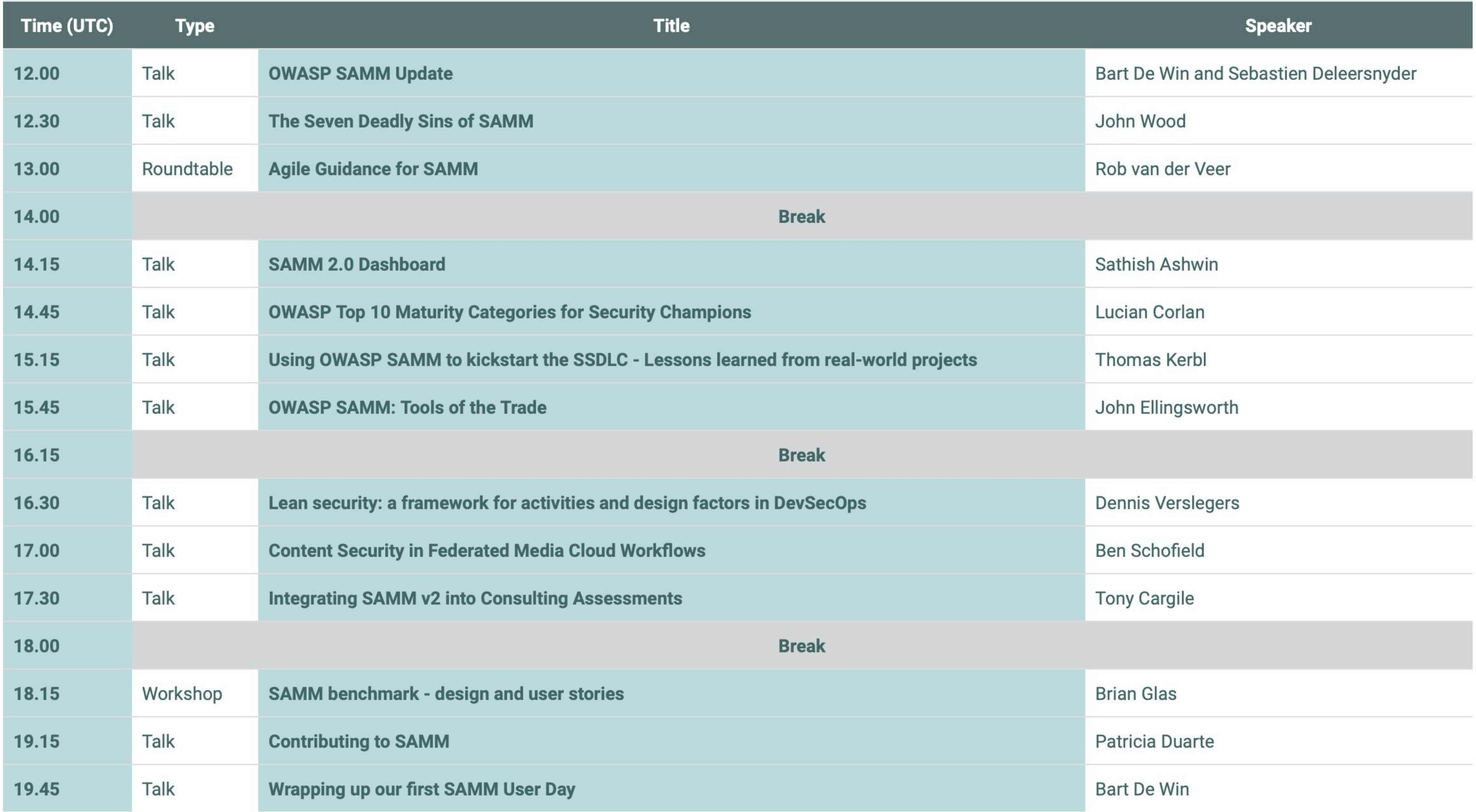
Bart De Win Project Co-Leader, Belgium	**Sebastien (Seba) Deleersnyder** Project Co-Leader, Belgium
Brian Glass – United States	Daniel Kefer – Germany
Yan Kravchenko – United States	Chris Cooper – United Kingdom
John DiLeo – New Zealand	Nessim Kisserli – Belgium
Patricia Duarte – Uruguay	John Kennedy – Sweden
Hardik Parekh – United States	John Ellingsworth – United States
Sebastián Arriada – Argentina	Brett Crawley – United Kingdom

SAMM Sponsors



owaspsamm.org/sponsors

Enjoy the User Day !



Time (UTC)	Type	Title	Speaker
12.00	Talk	**OWASP SAMM Update**	Bart De Win and Sebastien Deleersnyder
12.30	Talk	**The Seven Deadly Sins of SAMM**	John Wood
13.00	Roundtable	**Agile Guidance for SAMM**	Rob van der Veer
14.00		**Break**	
14.15	Talk	**SAMM 2.0 Dashboard**	Sathish Ashwin
14.45	Talk	**OWASP Top 10 Maturity Categories for Security Champions**	Lucian Corlan
15.15	Talk	**Using OWASP SAMM to kickstart the SSDLC - Lessons learned from real-world projects**	Thomas Kerbl
15.45	Talk	**OWASP SAMM: Tools of the Trade**	John Ellingsworth
16.15		**Break**	
16.30	Talk	**Lean security: a framework for activities and design factors in DevSecOps**	Dennis Verslegers
17.00	Talk	**Content Security in Federated Media Cloud Workflows**	Ben Schofield
17.30	Talk	**Integrating SAMM v2 into Consulting Assessments**	Tony Cargile
18.00		**Break**	
18.15	Workshop	**SAMM benchmark - design and user stories**	Brian Glas
19.15	Talk	**Contributing to SAMM**	Patricia Duarte
19.45	Talk	**Wrapping up our first SAMM User Day**	Bart De Win

OWASP SAMM outreach 2020

Using our social media, sponsor and subscriber networks:

- Twitter - 900 followers
- LinkedIn - 120 followers
- Newsletter - 600+ subscribers
- OWASP SAMM - 8 Sponsors
- SAMM Slack channel - 400+ members

OWASP
Open Web Application
Security Project

OWASP SAMM

Thank you!