



Secure SDLC für die Masse dank OpenSAMM?

Dr. Bruce Sams

OWASP

17.11.2011

Copyright © The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

The OWASP Foundation

<http://www.owasp.org>

Sicherer Software Development Lifecycle

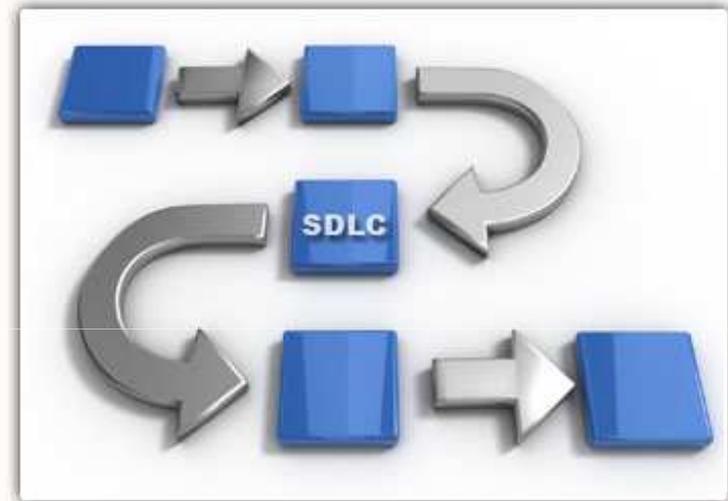
Weg von punktuellen Maßnahmen hin zur strategischen Planung

Der sichere SDLC (SLC?) soll Ordnung aus dem Chaos bringen

Erst jetzt kommen brauchbare Standards wie OpenSAMM an.

Es gibt viele offene Fragen, z.B:

- Welche Maßnahmen gehören dazu?
- Prozess- oder Maturity-Modell?
- Allgemeine Anwendbarkeit?



Die Historie des sicheren SDLCs

Name	Jahr	Merkmale
TSP-Secure	?	Fokus auf "defect removal", eigenständige Teams
CMMI	2002	Für allgemeine Entwicklung, kein Fokus auf Sicherheit, Reifegradmodell
Microsoft SDL (Prozess)	2004	Prozess, sehr stark integriert, speziell auf Microsoft-ähnliche Organisationen angepasst
OPTIMAbit Secure SDLC	2004	Prozessorientiert mit Touchpoints
CLASP (Prozess)	2005	Lose Sammlung von Prozesserweiterungen, Tools, Vulnerabilitykategorien etc.
Touchpoints	2006	Prozesserweiterungen ähnlich, CLASP aber strukturierter
OpenSAMM (Software Assurance Maturity Model)	2008	Reifegradmodell, inkrementell, anpassbar, basiert auf Expertenmeinung, detailliert
BSI-MM (Build Security In Maturity Model)	2009	Reifegradmodell, inkrementell, anpassbar, basiert auf Studie, Übersetzung auf Deutsch!

Prozess- vs. Reifegrad-Modelle

Erfolgsversprechend sind nur Reifegradmodelle

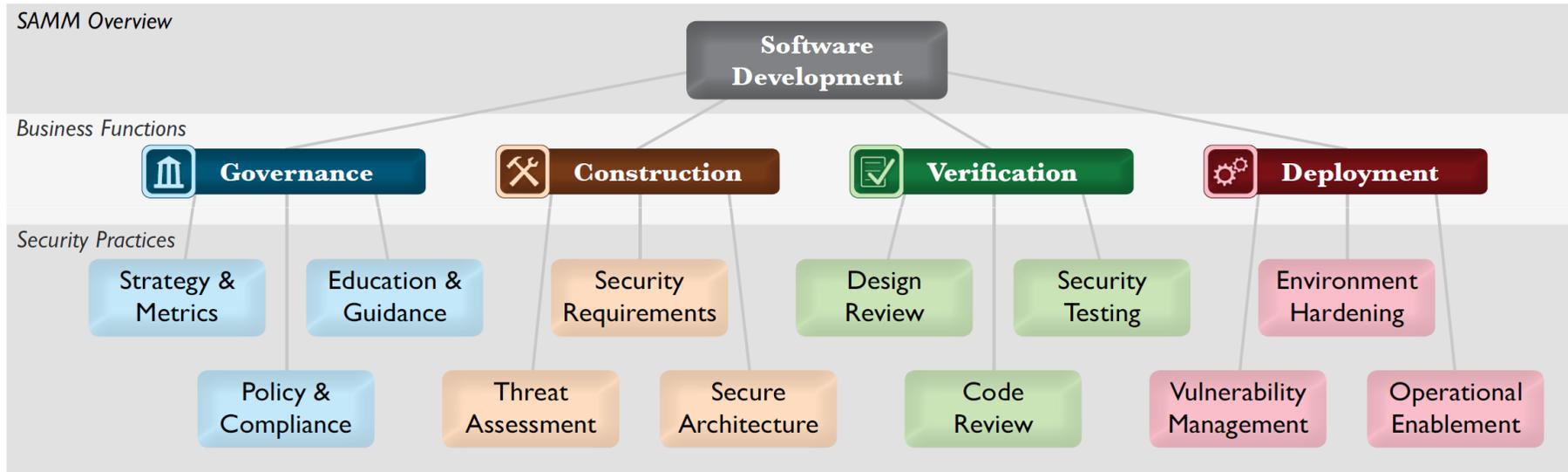
Softwareentwicklung ist vielfältig und jedes Unternehmen hat seine eigene Prozesse und Verfahren dazu.

Nur ein Reifegradmodell, welches auf eine höhere Ebene agiert, kann die Vielfalt und die Verwandlung der Softwareentwicklung abdecken.

Insource	vs	Outsource
Formal	vs	Agile
Build	vs	Buy
Streng	vs	Anything Goes

Überblick OpenSAMM

4 Geschäftsbereiche, 12 Bereiche der Sicherheitspraktiken



OpenSAMM Reifegrad

Level 3

- Meisterschaft
- Fortgeschrittene Maßnahmen
- Strukturierte Prozesse, Controls

Level 2

- Erhöhte Effizienz
- Gute Maßnahmen
- Einfache Prozesse

Level 1

- Erstes Verständnis
- Basis Maßnahmen
- Keine Prozesse

Level 0

- Impliziter Startpunkt
- Keine/wenige Maßnahmen
- Ad-hoc Implementierung



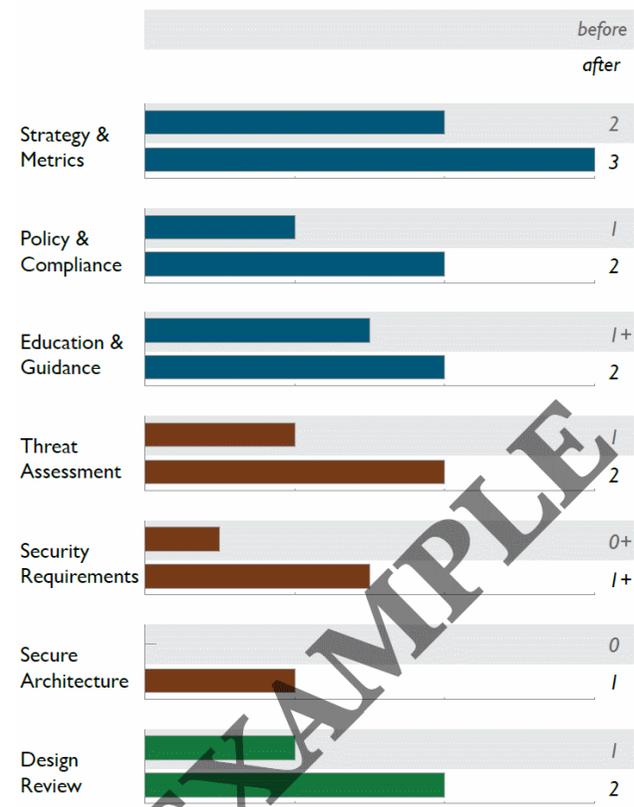
Beispiel Inhalt

Code Review ...more on page 62		
 CR 1	 CR 2	 CR 3
Opportunistically find basic code-level vulnerabilities and other high-risk security issues	Make code review during development more accurate and efficient through automation	Mandate comprehensive code review process to discover language-level and application-specific risks
<ul style="list-style-type: none">A. Create review checklists from known security requirementsB. Perform point-review of high-risk code	<ul style="list-style-type: none">A. Utilize automated code analysis toolsB. Integrate code analysis into development process	<ul style="list-style-type: none">A. Customize code analysis for application-specific concernsB. Establish release gates for code review

Assessment & Scorecards

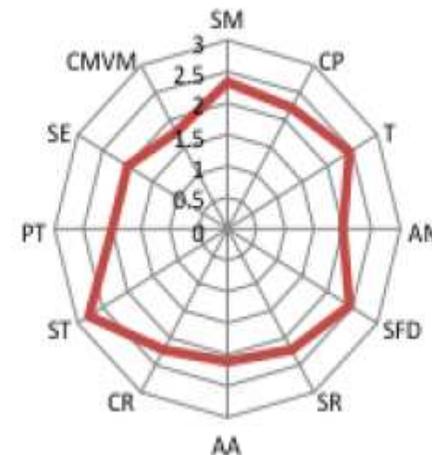
Scorecards zeigen den aktuellen/alten Stand

- Es existiert einen Satz an Fragen, die auf die Aktivitäten zielt.
- So ermittelt man den aktuellen Stand eines Unternehmens



Values for active companies

- BSIMM Survey
- Neun Top-Unternehmen aus Finanz, Web und Software wurden befragt.
- Typische Ergebnisse sind Level 2.
- Werte nach „High Water Mark“



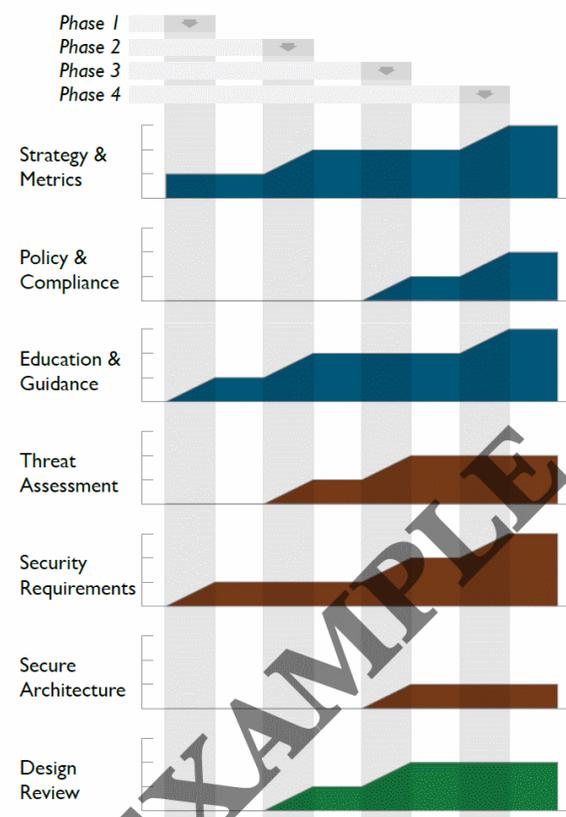
Beispielfragen zum Reifegrad

VERIFICATION	
Security Testing	
ST 1	ST1.1 Are projects specifying some security tests based on requirements?
	ST1.2 Do most projects perform penetration tests prior to release?
	ST1.3 Are most stakeholders aware of the security test status prior to release?
ST 2	ST2.1 Are projects using automation to evaluate security test cases?
	ST2.2 Do most projects follow a consistent process to evaluate and report on security tests to stakeholders?
ST 3	ST3.1 Are security test cases comprehensively generated for application-specific logic?
	ST3.2 Do routine project audits demand minimum standard results from security testing?

Roadmaps

Roadmaps zeigen den Implementierungsplan

- Welches Level soll bis wann erreicht werden?
- Nicht alle Praktiken müssen auf Level 3 gebracht werden
- Vorgefertigte Roadmaps existieren



Threat Assessment

Threat Assessment ...more on page 46		
 TA 1	 TA 2	 TA 3
Identify and understand high-level threats to the organization and individual projects	Increase accuracy of threat assessment and improve granularity of per-project understanding	Concretely tie compensating controls to each threat against internal and third-party software
<ul style="list-style-type: none">A. Build and maintain application-specific threat modelsB. Develop attacker profile from software architecture	<ul style="list-style-type: none">A. Build and maintain abuse-case models per projectB. Adopt a weighting system for measurement of threats	<ul style="list-style-type: none">A. Explicitly evaluate risk from third-party componentsB. Elaborate threat models with compensating controls

Erfolgsmetriken

Metriken messen den Erfolg

- Metriken sind für Prozesse potentiell gefährlich.
- Die in OpenSAMM angegebene Metriken sind u.U. schwer erreichbar
- Realistische Ziele setzen (OpenSAMM ist etwas optimistisch)

ADD'L SUCCESS METRICS

- ◆ >90% applications and data assets evaluated for risk classification in past 12 months
- ◆ >80% of staff briefed on relevant application and data risk ratings in past 6 months
- ◆ >80% of staff briefed on relevant assurance program roadmap in past 3 months



Kostenmodell

Kosten werden schätzungsweise angegeben

- Kostenschätzung ist vielfältig
- OPTIMAbit hat ein Modell für die Kostenschätzung entwickelt mit Kostenverteilung auf
 - ▶ Startup
 - ▶ pro Projekt
 - ▶ laufende

ADD'L COSTS

- ◆ Buildout or license of application and data risk categorization scheme
- ◆ Program overhead from more granular roadmap planning

ADD'L PERSONNEL

- ◆ Architects (2 days/yr)
- ◆ Managers (2 days/yr)
- ◆ Business Owners (2 days/yr)
- ◆ Security Auditor (2 days/yr)

Mitarbeit am Standard

Eine Evaluierung vom OpenSAMM für Webapps

- Bachelorarbeit durch TUM und Fraunhofer SIT
- Vorteile/Nachteile OpenSAMM im Umfeld von Webanwendungen
- Wir könnten einige Verbesserungen vorschlagen (Klarifikation Pentest, Netzwerksicherheit, Highlevel Policy)
- Eine neue Version ist in der Arbeit



Technische Universität München

Fakultät für Informatik

Bachelorarbeit in Informatik

Eine Evaluierung von OpenSAMM
für die Entwicklung sicherer
Webanwendungen

Fabian Streitl

Vergleich SDLCs

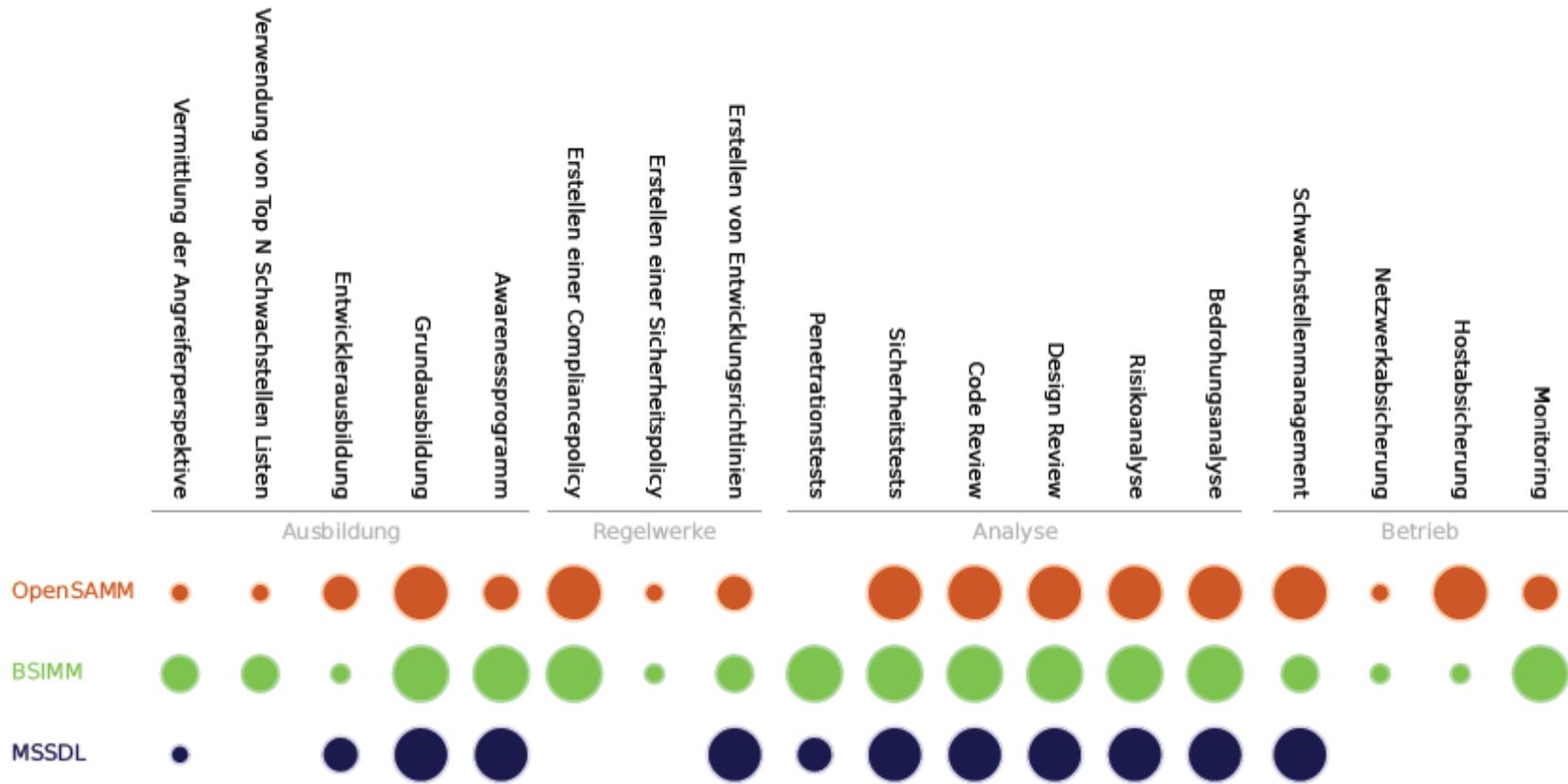
OpenSAMM, BSIMM und MSSDL

Aufteilung in gemeinsame Aktivitätengruppen

- Ausbildung
- Regelwerke
- Analyse
- Betrieb

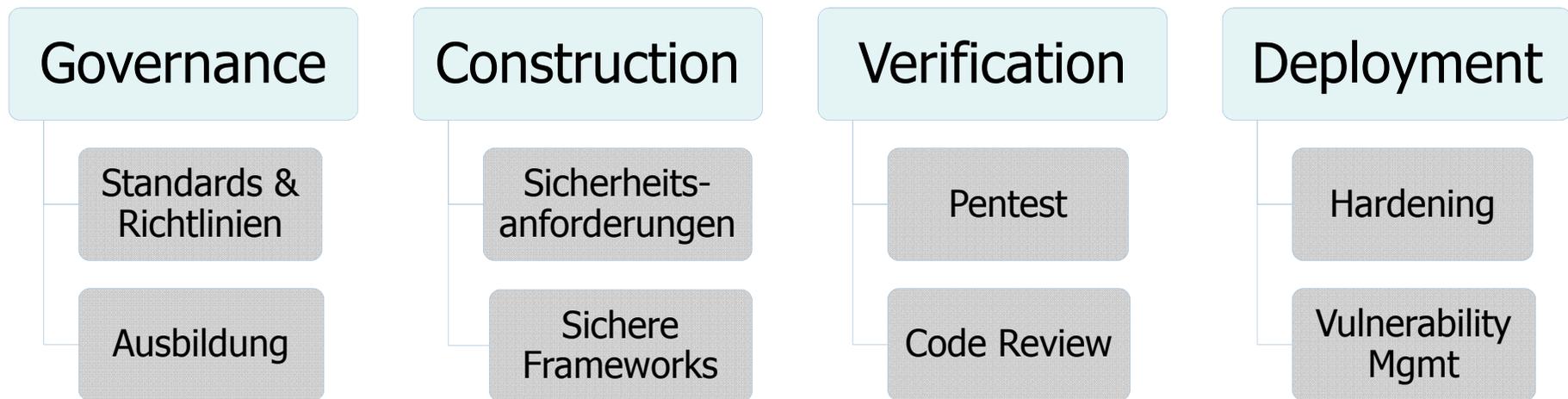


Vergleich SDLC Modelle



Wichtige Bestandteile eines secure SDLC

Diese Aktivitäten setzen eine Basislinie für sichere Anwendungen



Zusammenfassung

Mehrere konkrete Schritte führen zu einem sicheren SDLC

- Regelmäßiger Code Review
- Strukturierter Penetrationstest
- Ausgereifte Standards mit Checklisten
- Training und Awareness
- Verwendung sichere Frameworks
- Und mehr...

Kontakt

OPTIMAbit GmbH
Dr. Bruce Sams
Marktplatz 2
85375 Neufahrn

Tel.: +49 8165/65095
Fax +49 8165/65096

bruce.sams@optimabit.com
www.optimabit.com