Security Research Labs

Marc Heuse
Code Assurance Lead

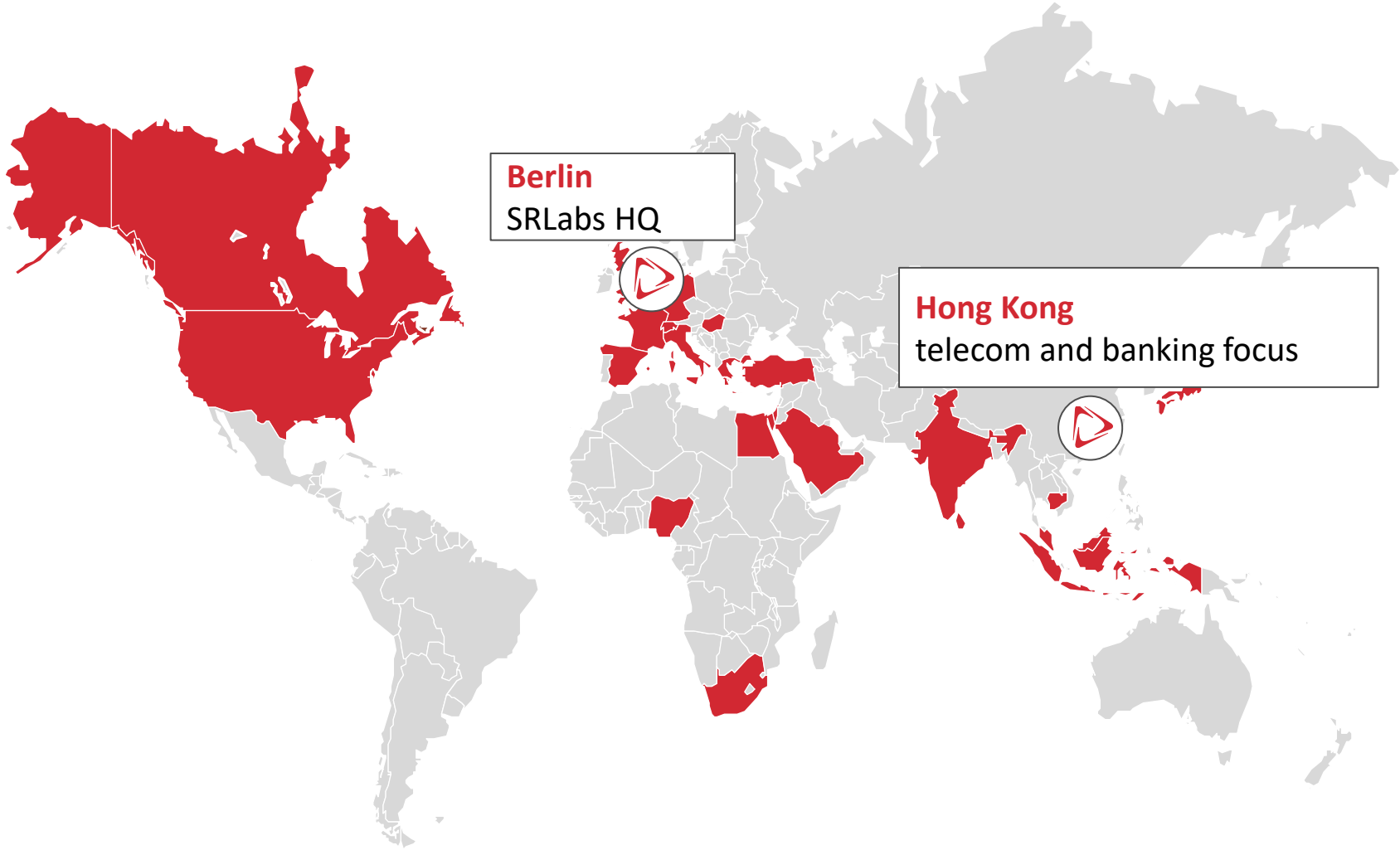
Sichere Anwendungsentwicklung für die Zukunft

öffentlich

Willkommen zu meiner Keynote!

Marc “vanHauser” Heuse

- 30 Jahre IT-Sicherheits-Know-how
- Autor mehrerer Top-100 Tools: AFL++ Hydra, thc-ipv6, THC-Scan, SuSEfirewall*, usw.
- Fokus auf komplexe Softwarearchitekturen von Konzepten über Quellcode und Fuzzing bis hin zu Reverse Engineering



Sicherheit in agilen Entwicklungsmethoden zu integrieren **ist eine große Herausforderung**

$150m

Keine Sicherheit im agilen Entwicklungsprozess

$425m

Fehlerhaftes Supply-Chain Management

£20m

Fehlerhaftes Sicherheitstesting

Agile Softwareentwicklung ermöglicht **immer kürzere Veränderungsprozesse**

In uralten, längst vergessenen Zeiten nutzten wir das **Wasserfall-, Prince2-** oder **V-Modell** zur Softwareentwicklung

Veränderungsprozesse werden immer kürzer, daher **dominieren** inzwischen klar **agile Entwicklungsmodelle**

	Lineare Softwareentwicklung	**Agile Softwareentwicklung**
Anforderungs-management	**Feste** Anforderungen zu Beginn	**Kontinuierliche** Anforderungen
Auswirkungen von Änderungen	**Hoch** - Änderungen sind teuer, Überarbeitung in mehreren Phasen erforderlich	**Gering** - Änderungen können schneller integriert werden, Prozess ist iterativ und flexibel
Lieferzyklen	**Lange** Zyklen, ein großes Release	**Kurze** Zyklen, häufige Releases

Die meisten agilen Softwareprojekte **haben Qualitätsprobleme**

Die meisten Firmen berichten über **Qualitätsprobleme in ihrer Agilen Entwicklung**

48%

Technische Schulden & mangelnde Qualität

46%

Kulturelle Widerstände

43%

Mangelnde Fähigkeiten & Erfahrung

41%

Fehlende Management-unterstützung

https://www.teculture.com/post/2020-14th-annual-state-of-agile-report-by-versionone-inc

Feature-getriebene Entwicklung begünstigt **Vernachlässigungen in der Wartbarkeit**

Kultur und Prozesse	1	*Agile fördert eine auf sofortige Anforderungen ausgerichtete Entwicklung*
Externe Einflüsse	2	*Stakeholder drängen auf ständige Feature-Updates*
Technische Schulden	3	*Wartung wird vernachlässigt, Pflegeaufwand steigt*

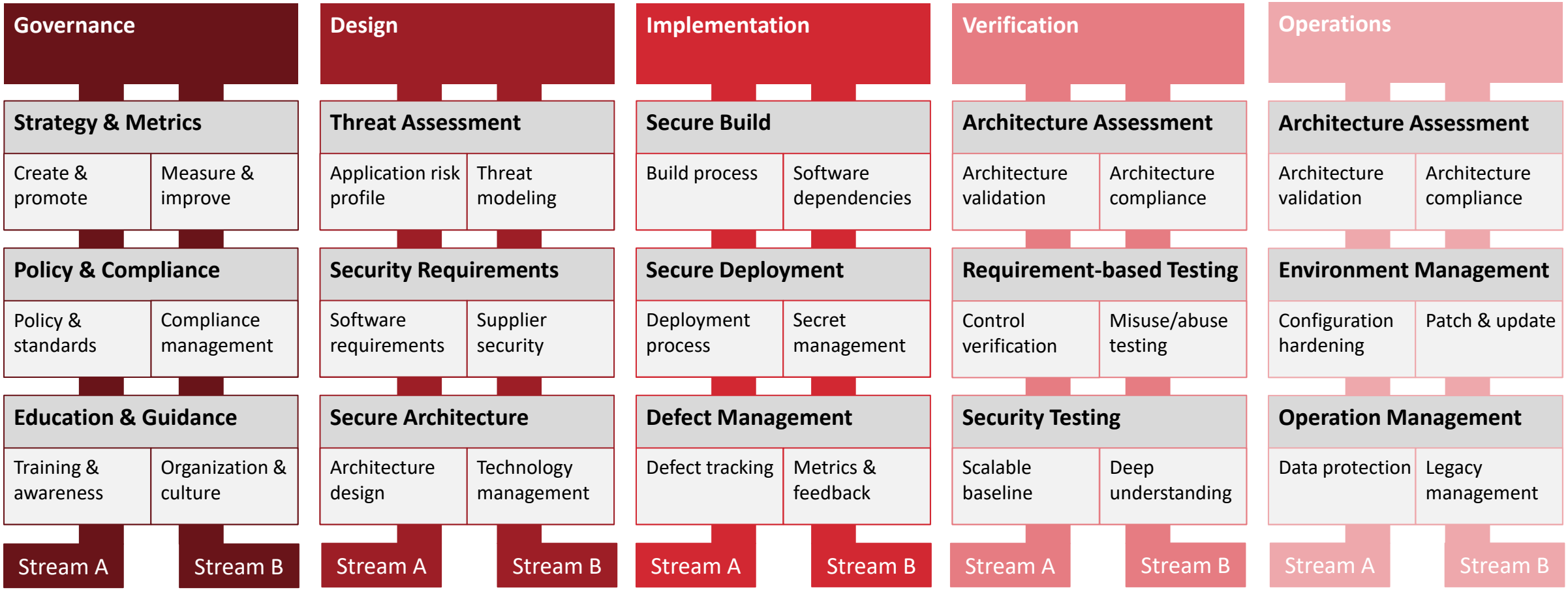
Wenn ein Haus **agil gebaut werden würde**

Sicherheit **integriert sich nicht von selbst** in den Softwareentwicklungsprozess

Agile Softwareentwicklung wird selten korrekt umgesetzt - doch erst dann kann “Sicherheit” integriert werden

Kernelemente erfolgreicher agiler Entwicklung	
Kulturelle Veränderungen und Schulungen	**Prozess wird zum Erfolg**, Team Motivation
Kontinuierliche Integration und Refactoring	**Gute Performance und Wartbarkeit**, niedrige Fehlerraten
Fokus auf Qualität und technische Exzellenz	**Zufriedene Kunden & Teams**

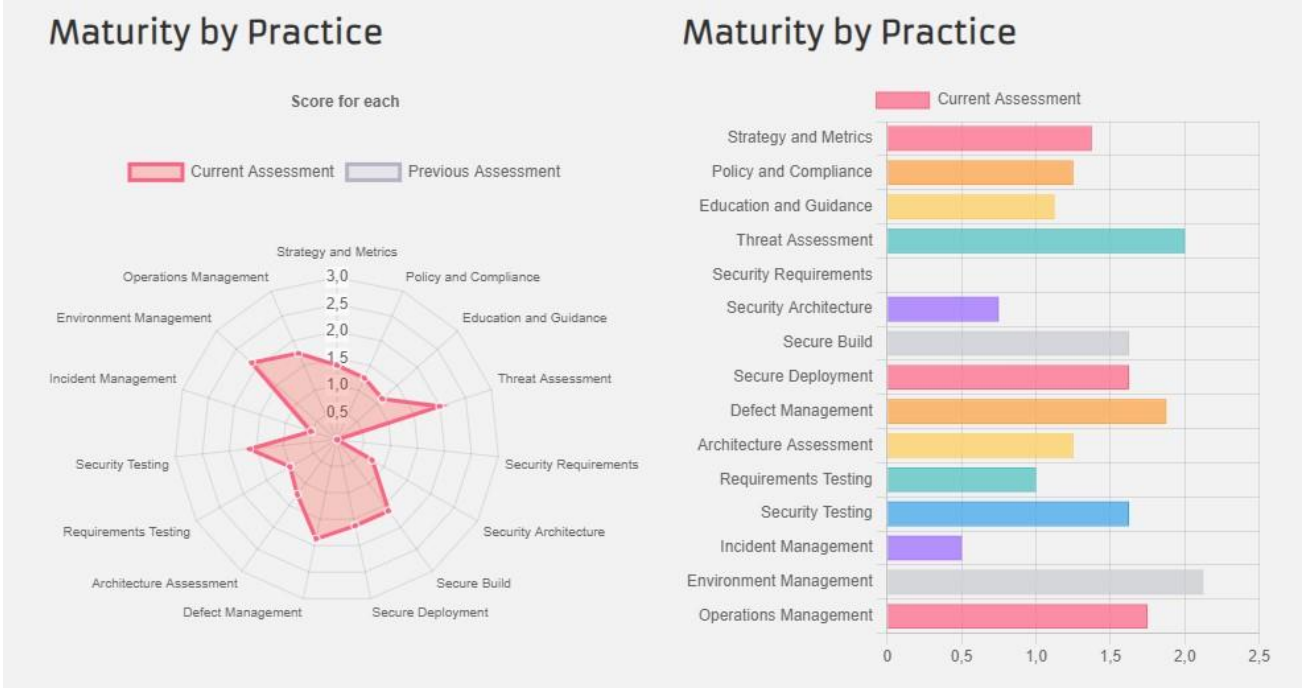
Die Integration von Sicherheit in den Softwareentwicklungs-prozess **ist eine komplexe Herausforderung**



https://owaspsamm.org/

Um sicherzustellen, dass das sicherheitsorientierte Entwicklungsmodell funktioniert, **sind Messungen erforderlich**

SAMMwise



https://owaspsamm.org/assessment/

“Hey, **wo ist die *‘Zukunft’***, die uns versprochen wurde?“

Die Zukunft **hält neue**
Herausforderungen für uns bereit
(während die alten noch ungelöst sind)

Zukunftsthema: **LLM/AI - sinnvoll**, aber nicht überall

Code Review Assistent	**Gibt gute Hinweise**
Fuzzing Harnesses erstellen	**Einfaches geht, Kontext fehlt**
Coding / Coding Assistent	**Lösung passt z.T. nicht zum Problem, APIs veraltet**
Code Kommentieren	**Zu viel/zu wenig, Kontext fehlt**
Eigenständige Code Analyse	**Keine Datenflußanalyse**

Zukunftsthema: **Supplychain - ein kritischer Angriffsvektor** der kaum in den Griff zu bekommen ist

Bibliotheken ungepflegt/veraltet	**Risikoanalyse bei der Auswahl**
Updates nicht bekannt oder werden nicht eingespielt	**Prozesse für Überwachung und Einspielen von Updates etablieren**
Trojanisiert: kompromittiert Entwickler, Buildsystem und/oder Anwender	… gibt es bis jetzt keine Lösung ☹

Seien Sie bereit für die Zukunft,
indem Sie **die Probleme von heute lösen**

So kommen Sie zum Ziel

1. Implementieren	2. Prüfen	3. Sicherheit	4. Messen
Setzen Sie den agilen Entwicklungsprozess korrekt auf und leben Sie ihn	**Prüfen Sie**, dass der agile Entwicklungs-prozess wirklich umgesetzt wird	**Integrieren Sie OWASP SAMM** in den Prozess	**Führen Sie ein Messsystem ein**, messen Sie kontinuierlich und **handeln Sie danach!**

Das könnten SIE sein!

agiles DevOps mit Chaos Engineering

- CI/CD + Sicherheitstests,
- Threat Modeling,
- regelmäßige Pentests

DevSecOps und autonome "Squads“

- Regelmäßige Pentests
- SecDev Trainings
- kontinuierliches Messen & Verbessern

Secure SAFe

- CI/CD + Sicherheitstests
- regelmäßige Pentests
- Bug-Bounty-Programm
- Threat Modeling

Danke für Ihre Aufmerksamkeit!

Security Research Labs

Marc Heuse
Code Assurance Lead

marc@srlabs.de
+49-(0)177-9611560