

Eine Einführung in das OWASP SAMM

sec4dev Dialogues 2024

whoami

- Mathias Tausig – mtausig@sba-research.org
- Technical IT Security Consultant at SBA Research
 - Penetration testing, SDLC, Threat Modeling, Cloud (Native) Security, ...
- Formerly SysAdmin, Developer, Security Officer, University teacher



When you are a hands-on guy and start consulting

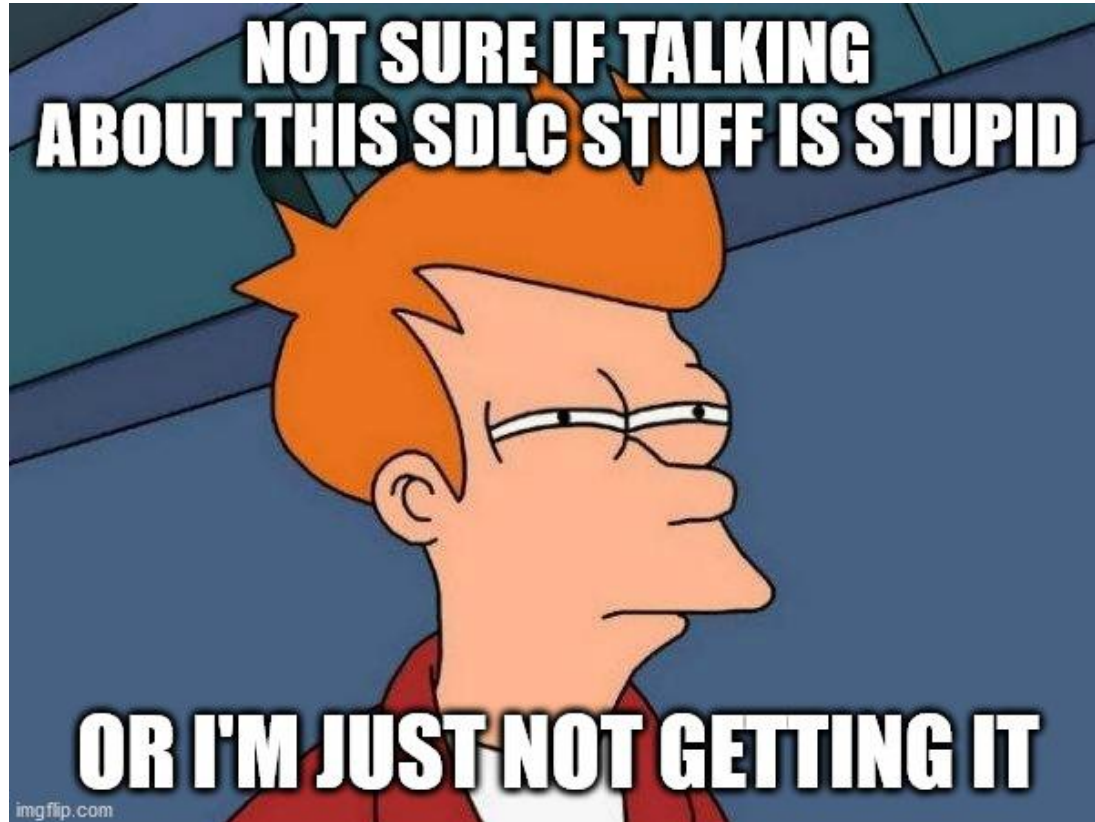


**SECURE
DEVELOPMENT
LIFECYCLE AUDITS**



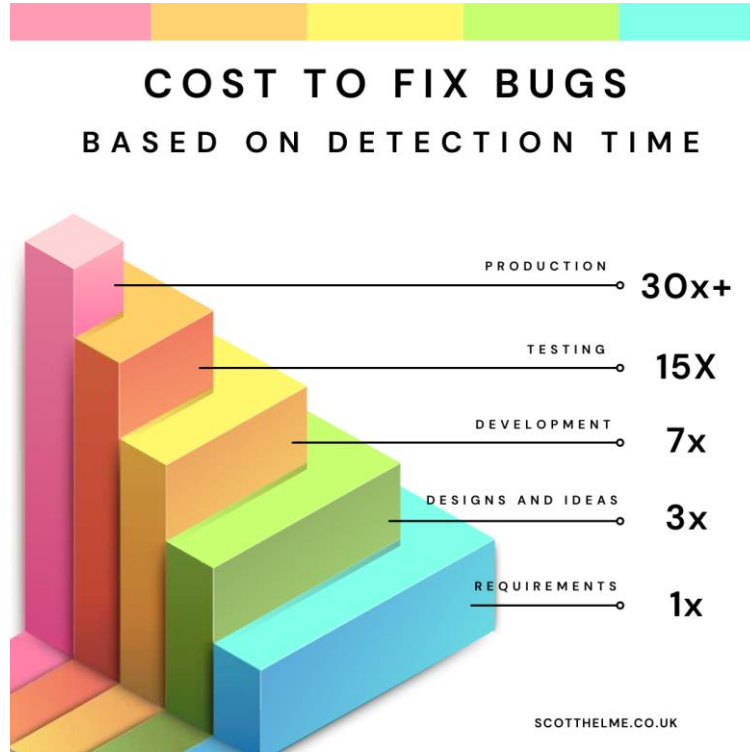
**WEBAPP
PENTEST**

When you start digging deeper ...





Shift Left



<https://scotthelme.co.uk/google-announce-new-minimum-viable-secure-product/>

Secure Development is not just Secure Coding

Example 1: Vulnerability in library

1. Undefined responsibilities between Dev & Ops
2. Missing automation
3. Software dependencies are not checked during build or deployment
4. A known vulnerability in a 3rd party library goes unnoticed
5. Internet-facing application gets exploited



CHALLENGE ACCEPTED

What's next

How secure is your software development process currently?

- Yes/No/Maybe/Very/Can't tell
- How can you tell?



The model and the assessment

OWASP SAMM

OWASP SAMM



OWASP SAMM

- **What is it?**

- Concise set of interview questions across all relevant domains
- Granular score in all areas
- Proposals & activities how you can improve

You talk to a team, SAMM tells you what to talk about.

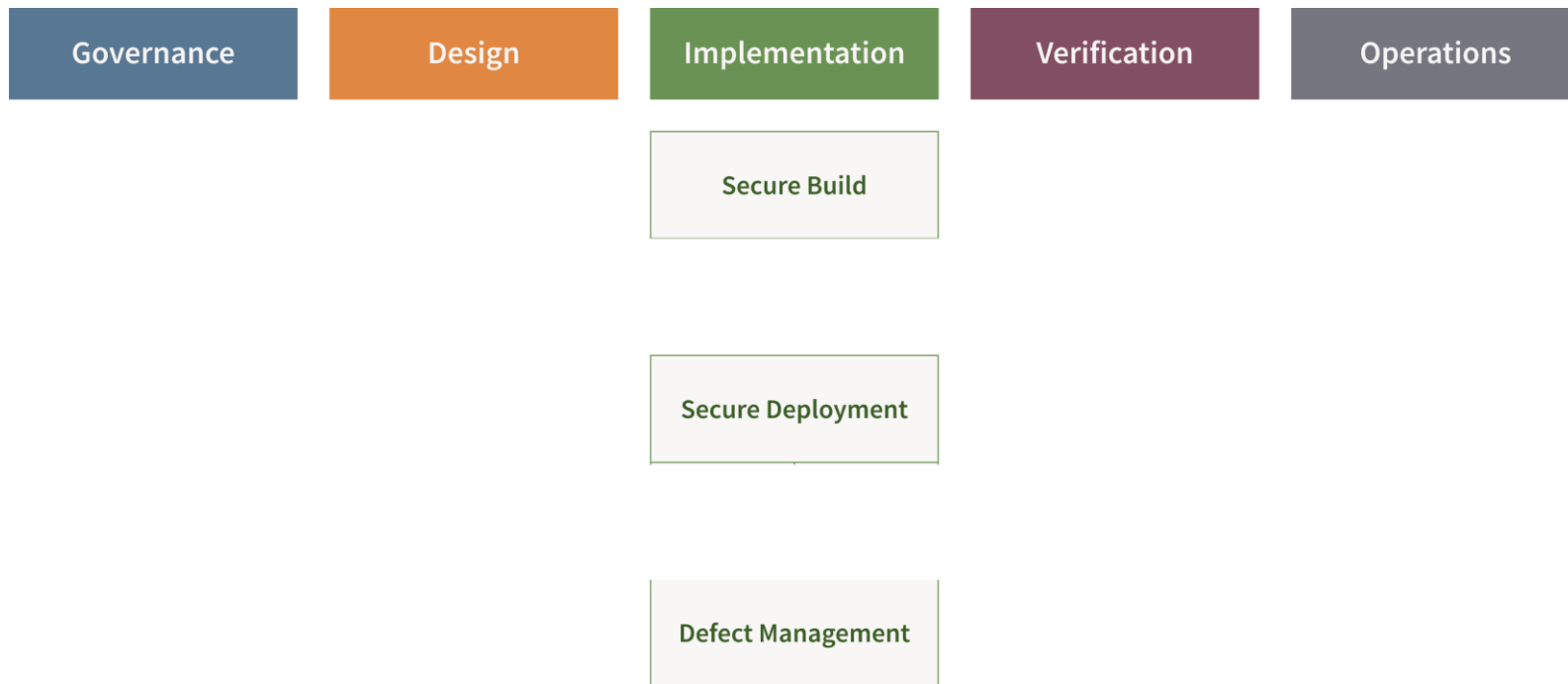
OWASP SAMM

Business functions



OWASP SAMM

Security practices



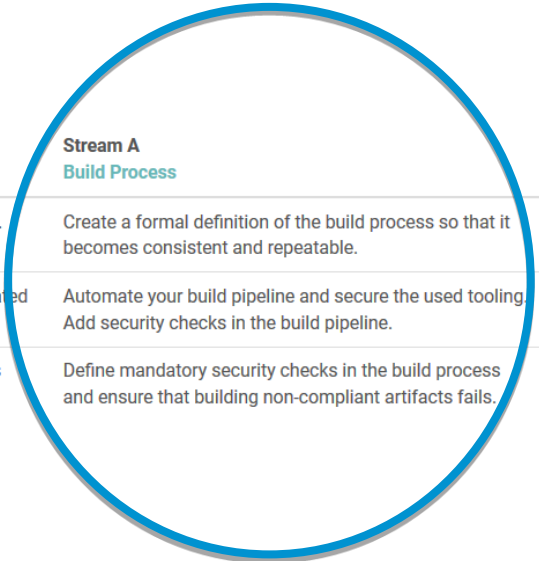
Stream / activity

Operations



OWASP SAMM

Maturity level



Maturity level		Stream A Build Process	Stream B Software Dependencies
1	Build process is repeatable and consistent.	Create a formal definition of the build process so that it becomes consistent and repeatable.	Create records with Bill of Materials of your applications and opportunistically analyze these.
2	Build process is optimized and fully integrated into the workflow.	Automate your build pipeline and secure the used tooling. Add security checks in the build pipeline.	Evaluate used dependencies and ensure timely reaction to situations posing risk to your applications.
3	Build process helps prevent known defects from entering the production environment.	Define mandatory security checks in the build process and ensure that building non-compliant artifacts fails.	Analyze used dependencies for security issues in a comparable way to your own code.

<https://owaspsamm.org/model/>

OWASP SAMM

Activities

Model | **Implementation** | **Secure Build** | **Build Process**

MATURITY LEVEL 1

MATURITY LEVEL 2

MATURITY LEVEL 3

Benefit

Limited risk of human error during build process minimizing security issues

Activity

Define the build process, breaking it down into a set of clear instructions to either be followed by a person or an automated tool. The build process definition describes the whole process end-to-end so that the person or tool can follow it consistently each time and produce the same result. The definition is stored centrally and accessible to any tools or people. Avoid storing multiple copies as they may become unaligned and outdated.

The process definition does not include any secrets (specifically considering those needed during the build process).

Review any build tools, ensuring that they are actively maintained by vendors and up-to-date with security patches. Harden each tool's configuration so that it is aligned with vendor guidelines and industry best practices.

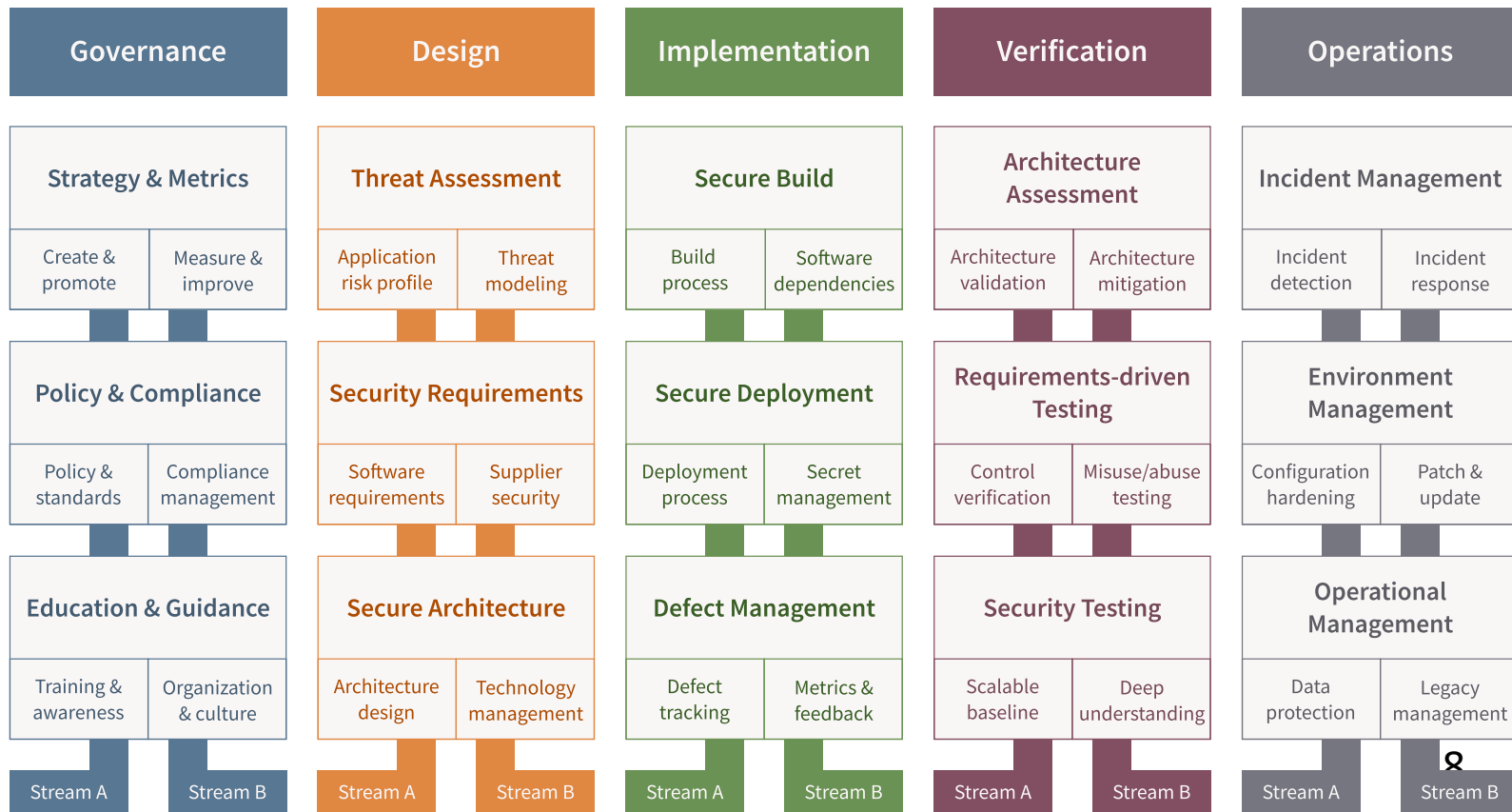
Determine a value for each generated artifact that can be later used to verify its integrity, such as a signature or a hash. Protect this value and, if the artifact is signed, the private signing certificate.

Ensure that build tools are routinely patched and properly hardened.

<https://owasp samm.org/model/implementation/secure-build/stream-a/>

OWASP SAMM

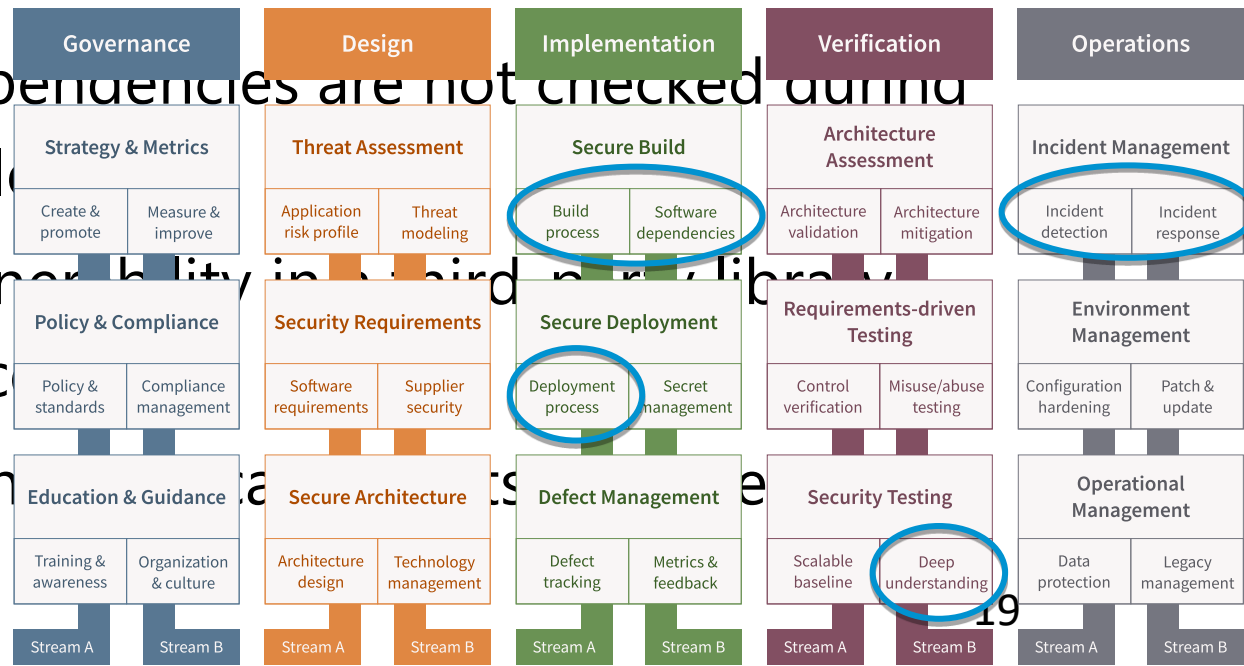
Overview



Secure Development is not just Secure Coding

Example 1, revisited

1. Undefined responsibilities between Dev & Ops
2. Missing automation
3. Software dependencies are not checked during build or deployment
4. A known vulnerability in a third-party library goes unnoticed
5. Internet-facing services are not properly secured



Output & Results

Scoring

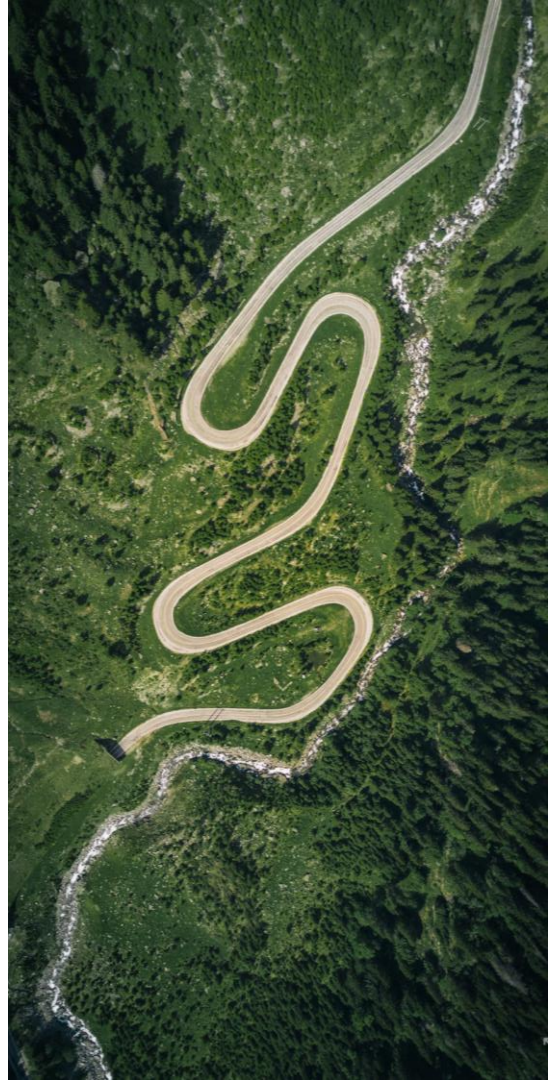
- **What you get**
 - A scored result for each function
 - Every activity has the same weight
 - Every level has the same weight
 - Helps detect blind spots
- **What you don't get**
 - Overall score

Current Maturity Score					
Functions	Security Practices	Current	Maturity		
			1	2	3
Governance	Strategy & Metrics	0,63	0,25	0,25	0,13
Governance	Policy & Compliance	0,63	0,50	0,13	0,00
Governance	Education & Guidance	0,75	0,38	0,13	0,25
Design	Threat Assessment	0,50	0,25	0,25	0,00
Design	Security Requirements	0,25	0,25	0,00	0,00
Design	Secure Architecture	0,88	0,50	0,13	0,25
Implementation	Secure Build	1,88	1,00	0,63	0,25
Implementation	Secure Deployment	1,13	0,75	0,38	0,00
Implementation	Defect Management	0,63	0,63	0,00	0,00
Verification	Architecture Assessment	0,88	0,75	0,00	0,13
Verification	Requirements Testing	0,75	0,25	0,25	0,25
Verification	Security Testing	1,50	0,75	0,50	0,25
Operations	Incident Management	0,13	0,13	0,00	0,00
Operations	Environment Management	0,50	0,38	0,13	0,00
Operations	Operational Management	1,25	1,00	0,13	0,13

Output & Results

Road map

- **Main output of assessment**
 - Status quo
 - Motivation and goals for short-term and long-term development
- **Where should I start?**
 - Ways to improve optimally and easiest
 - Activities that are almost established already
 - Most relevant activities in the given environment



Interviews

How assessments are done

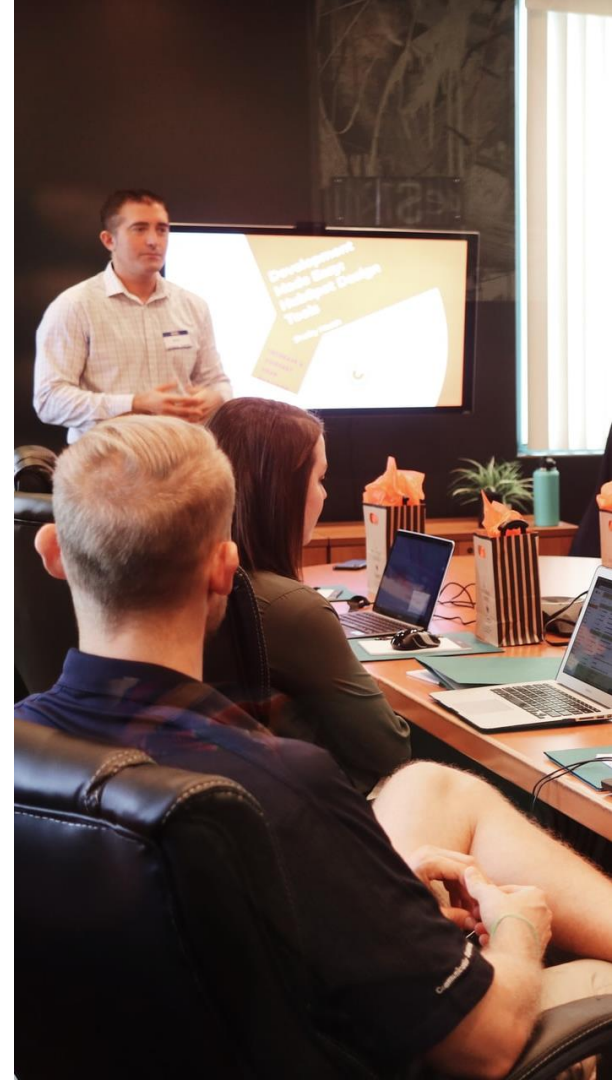
Assessment Types

- **External interviewers**
 - Security experts are interviewers
 - Report with suggestions for moving forward
- **Self assessment**
 - Interview done by the team itself
 - Can be done more often
 - Use external help to learn the process



Interview

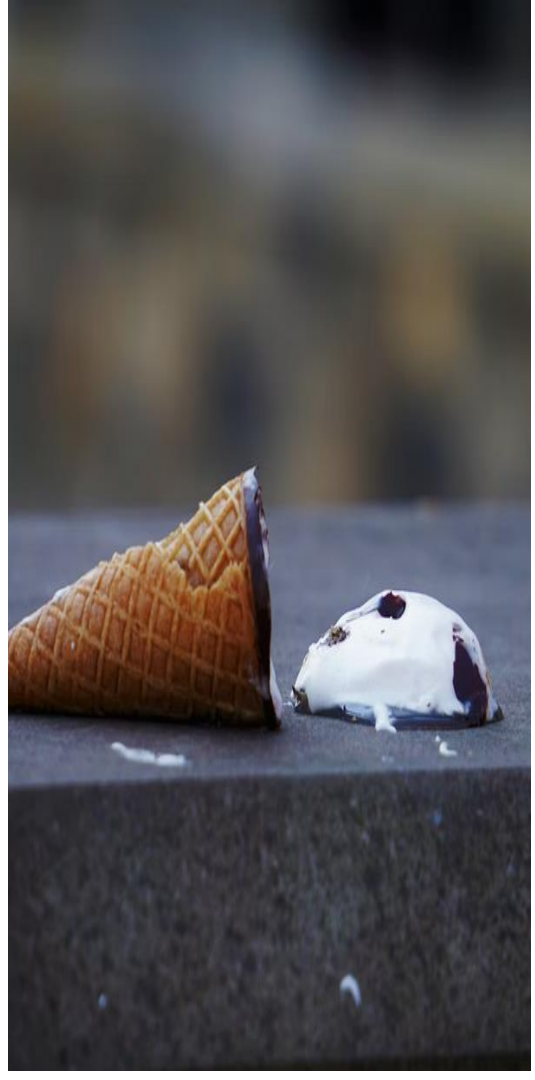
- 1-5 team members with different roles get interviewed
- 2 interviewers
- Preparation
 - Interviewers should know about team, organization & software
 - Teams should have relevant documents and software at hand
- Initially takes ~1 day to interview a team



Common mistakes #1

Does it make sense to compare teams by their SAMM score?

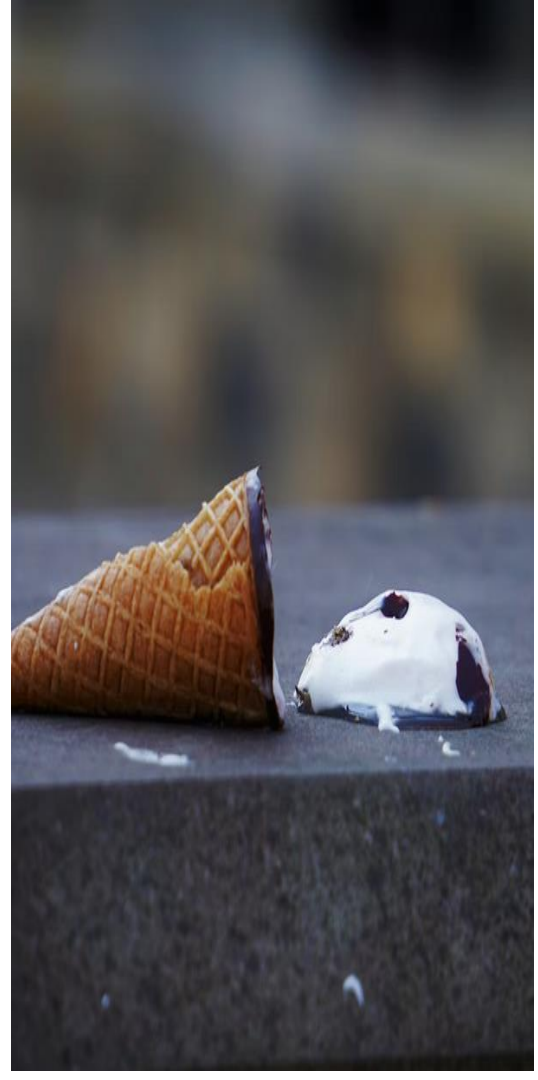
- No
- The only thing that matters, is the development of the score within a team
- Validate the progress
 - No fixed numbers or percentages; road map needs to be tailored to the team



Common mistakes #2

Should some absolute requirements be given to teams?

- No
- The process should spark intrinsic motivation for improvement instead



TL;DR

- **Invest in security early**
- Pick a project/team
- Conduct interview
 - Use [website](#) if you get lost
- Publish notes & scores
- Specify roadmap with easy wins & blind spots

