OWASP
Open Web Application
Security Project

OWASP
SAMM

A Complete View of Application Security with OWASP SAMM

Course contents

- The Application Security Challenge
- Software Development Lifecycle Overview
- OWASP SAMM
  - Vision, History, Structure
  - Assessment Tool
  - Deep Dive into Secure Build: Demo
  - Methodology
- Conclusion

SAMM Community

https://owaspsamm.org




OWASP SAMM free course

OWASP SAMM

Fundamentals Course

About this course

- Free
- 79 lessons
- 5 hours of video content

owaspsamm.thinkific.com/courses/samm

Learning Objectives & Expectations

- Understand the application security challenge
- Get a clear view of the AppSec landscape
- Learn about SAMM (i.e., the solution)

Aram Hovsepyan

- PhD @ DistriNet, KULeuven
- CEO @ Codific
- OWASP SAMM core team member

https://www.linkedin.com/in/aramhovsep
https://www.linkedin.com/company/codific

Terms of

reference

OWASP SAMM

What is security?

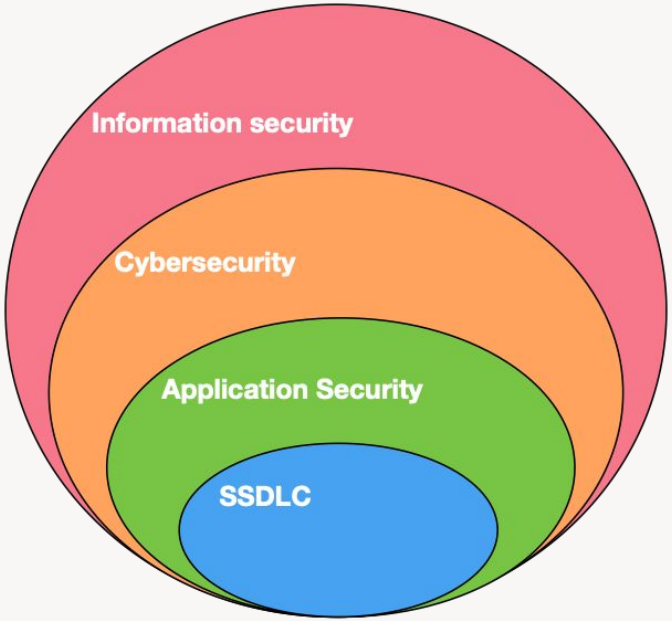
CONFIDENTIALITY
unauthorized users cannot access assets

INTEGRITY
unauthorized users cannot modify assets

AVAILABILITY
assets are available on request

The scopes for security

- SDLC
  - Build security in
- Application Security (AppSec)
  - Manage application security risks
- Cybersecurity
  - Protect digital assets
- Information Security (InfoSec)
  - Protect all assets



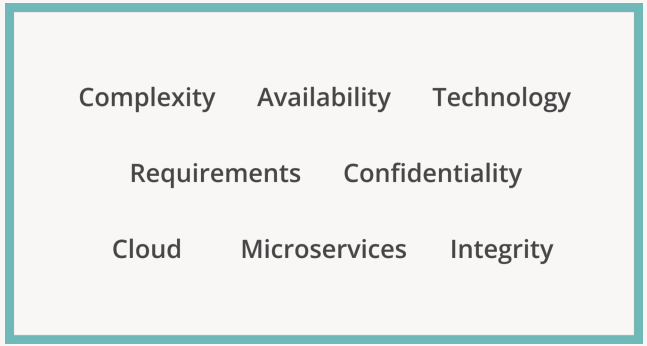
Framework / maturity model

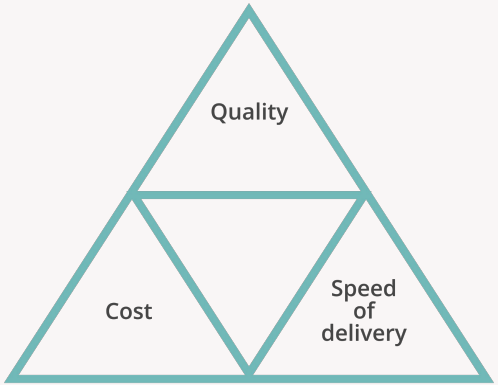
- ISO27001, NIST CSF, NIST SSDF, etc.
- Structured set of best practices
- Optional: methodology

The application security problem

OWASP SAMM

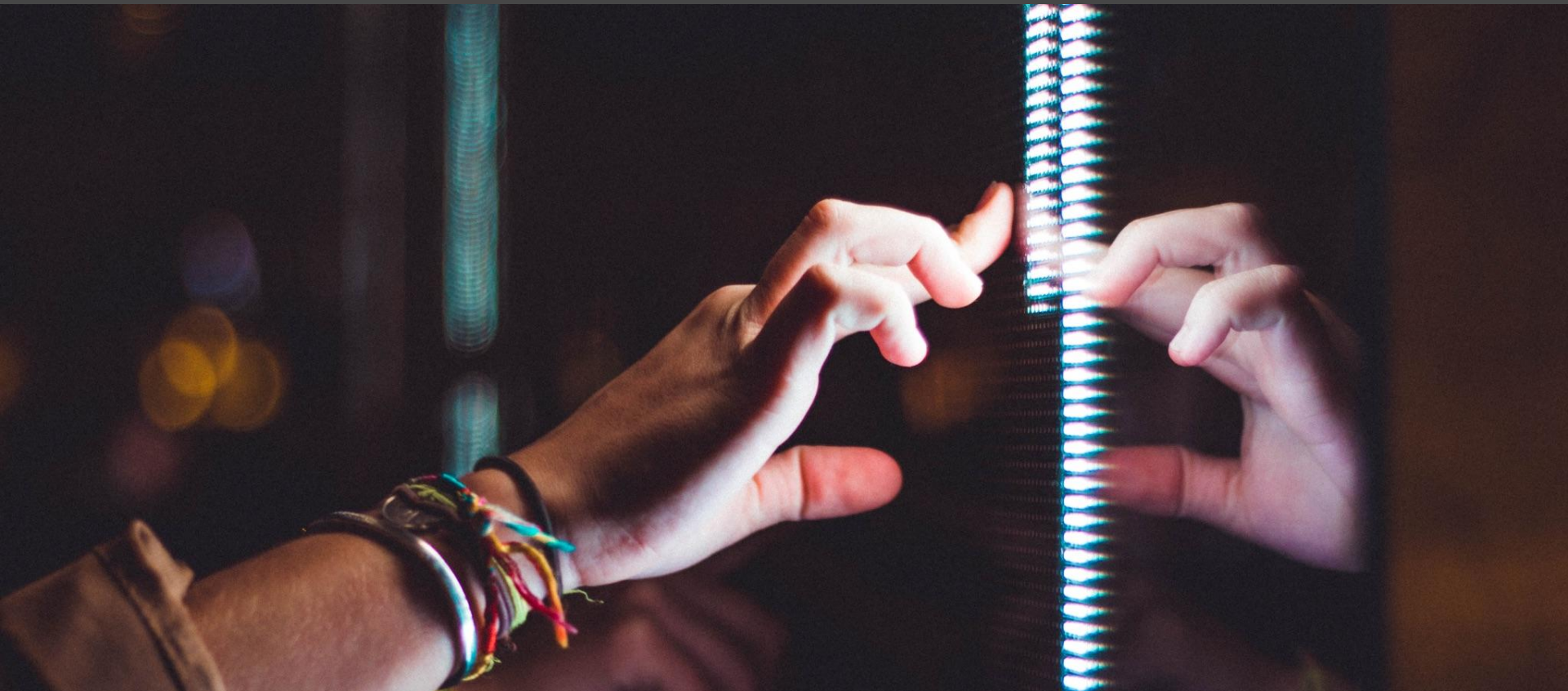
The application security problem





Application are the most common attack vector (DBIR'24)

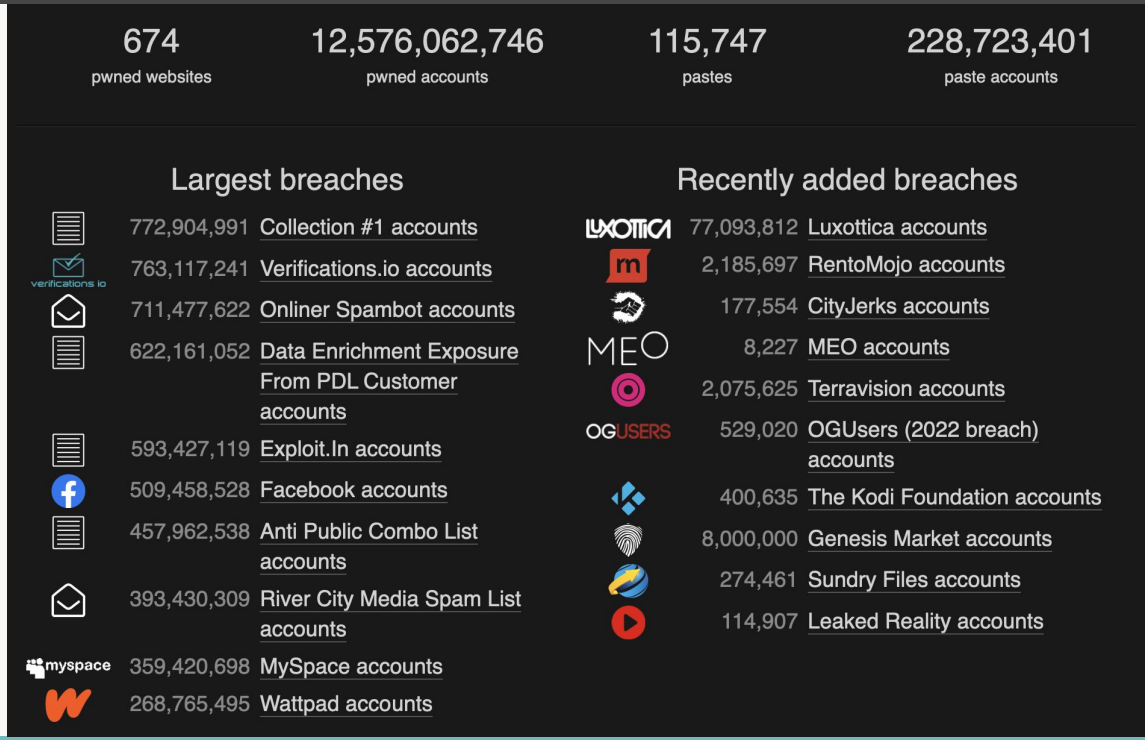
Security is intangible

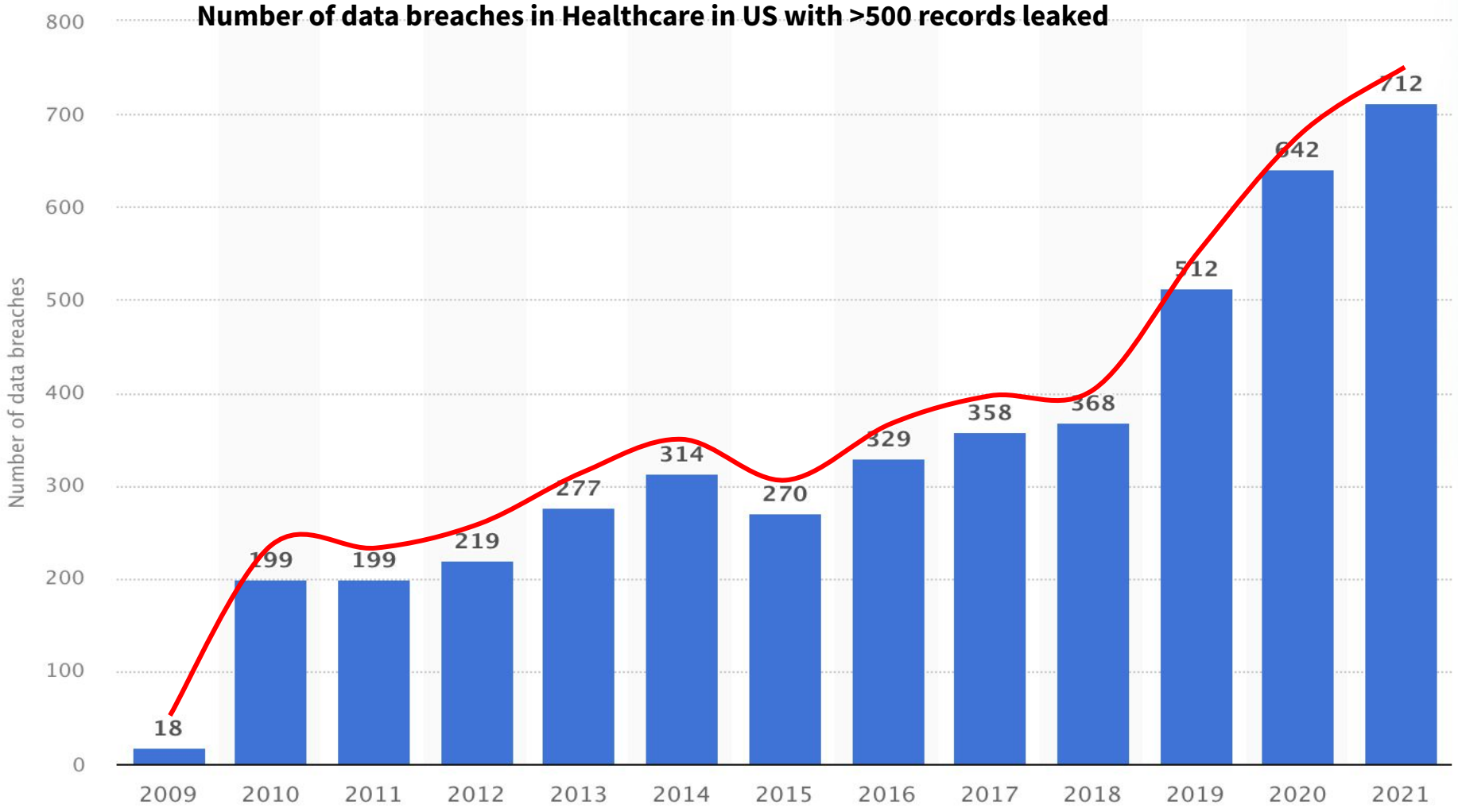


When do we feel (in)security?

Number of breaches is surging

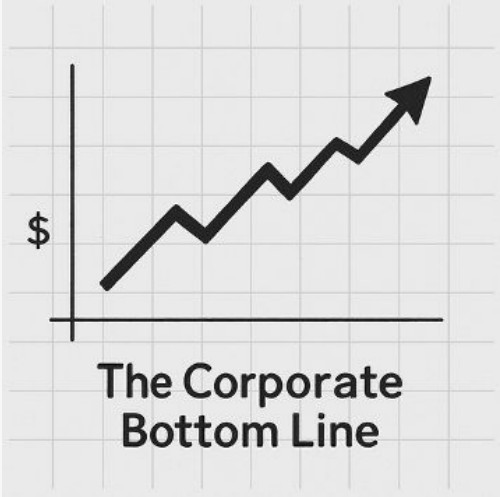
https://haveibeenpwned.com




Number of data breaches in Healthcare in US with >500 records leaked
Number of data breaches
800
700
600
500
400
300
200
100
0
18
199
199
219
277
314
270
329
358
368
512
642
712
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021

Data breach impact

- Fixing the issue
- Direct fines
- Loss of trust
- Reputational damage
- Stock price (*)
- Compensation requested by users

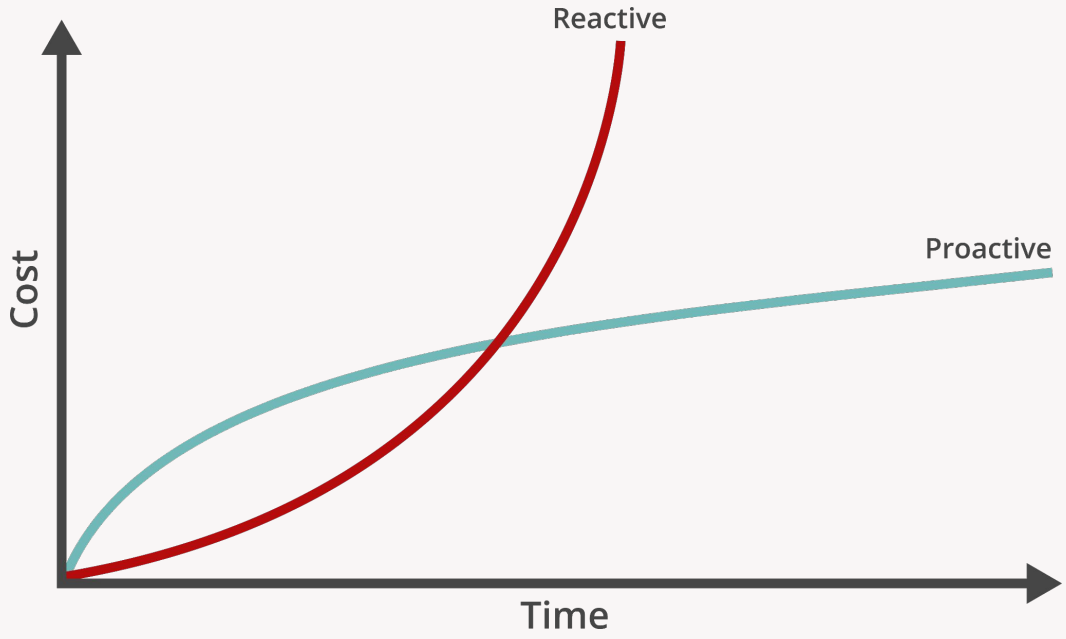


Security in the

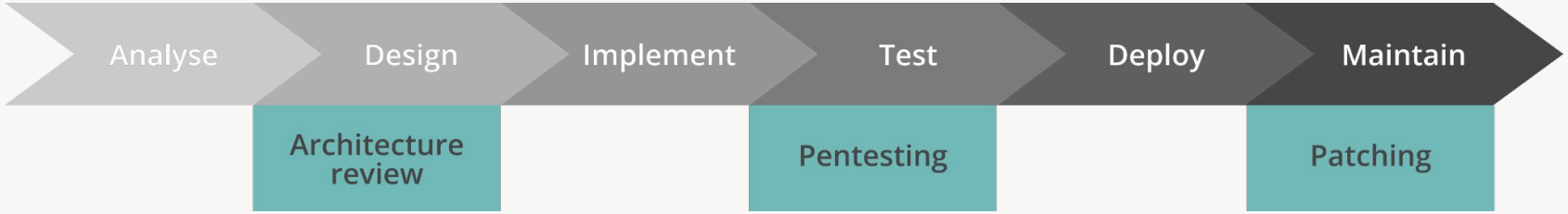
SDLC

OWASP SAMM

The cost of application security



Security in a traditional SDLC



Why is this problematic?

- It's not cost efficient
- There is no security assurance

Security in a traditional SDLC

https://nakedsecurity.sophos.com/2022/06/24/openssl-issues-a-bugfix-for-the-previous-bugfix/

Security in a traditional SDLC

Google: Half of 2022's Zero-Days Are Variants of Previous Vulnerabilities

Google Project Zero has observed a total of 18 exploited zero-day vulnerabilities in the first half of 2022, at least half of which exist because previous bugs were not properly addressed.

https://www.securityweek.com/google-half-2022s-zero-days-are-variants-previous-vulnerabilities/

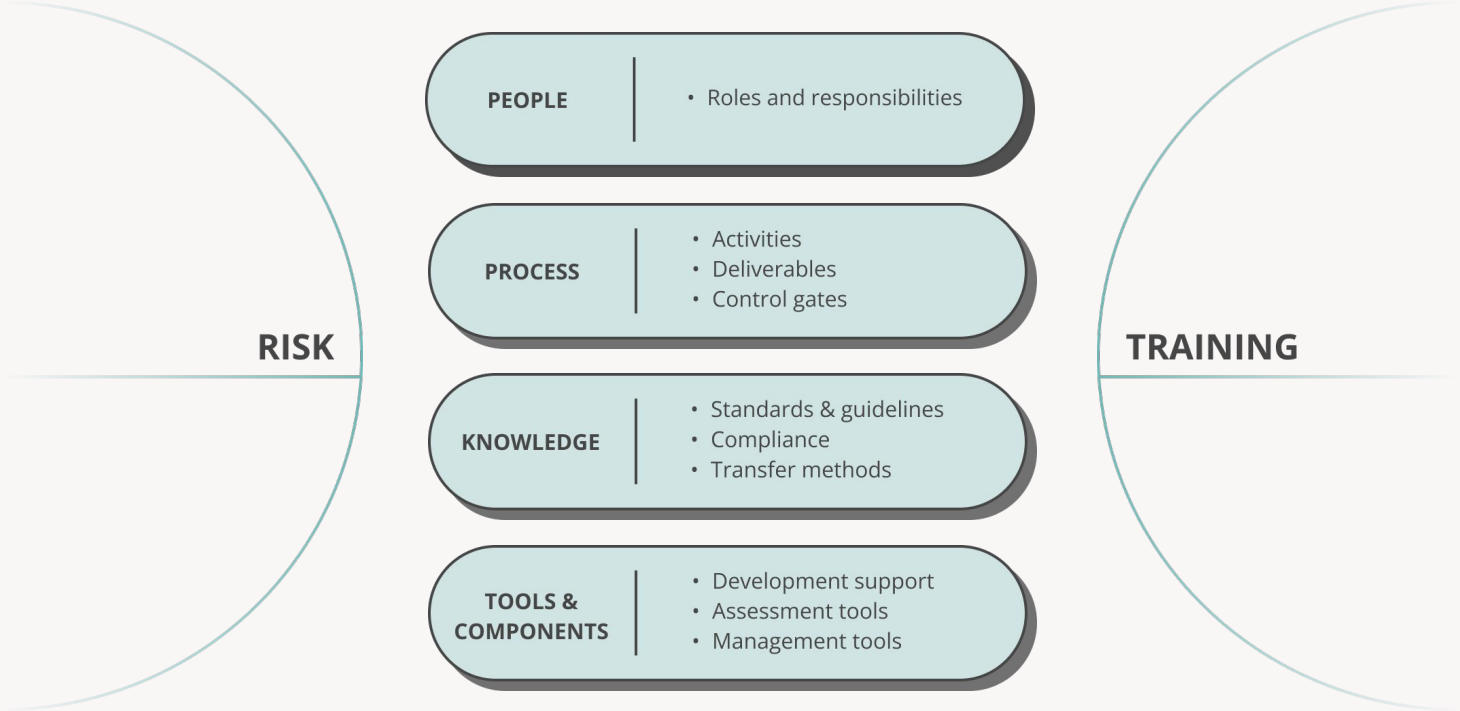
Secure by Design

Enterprise-wide software security improvement program

- Strategic approach to assure software quality
- Consistent and systematic approach

AppSec Initiatives

AppSec Cornerstones



Vision and history

OWASP SAMM

What is OWASP?

Open Worldwide Application Security Project

FLAGSHIP

mature projects

What is SAMM?

Software Assurance Maturity Model

Measurable
Defined maturity levels across business practices

Actionable
Clear pathways for improving maturity levels

Versatile
Technology, process, and organization agnostic

Why a maturity model?

An organization's behavior changes slowly over time

Changes must be **iterative** while working toward long-term goals

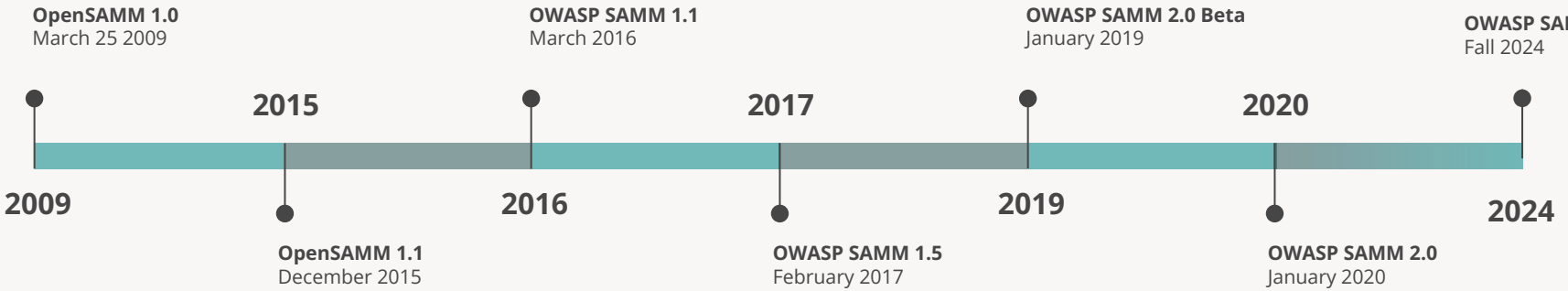
There is no single recipe that works for all organizations

A solution must enable **risk-based** choices tailored to the organization

Guidance related to security activities must be prescriptive

A solution must provide enough **details** for non-security-people

SAMM project history



Who is SAMM? Core team

Sebastien Deleersnynder
Bart De Win
Maxim Baele
Aram Hovsepyan

Daniel Kefer

John DiLeo

Patricia Duarte

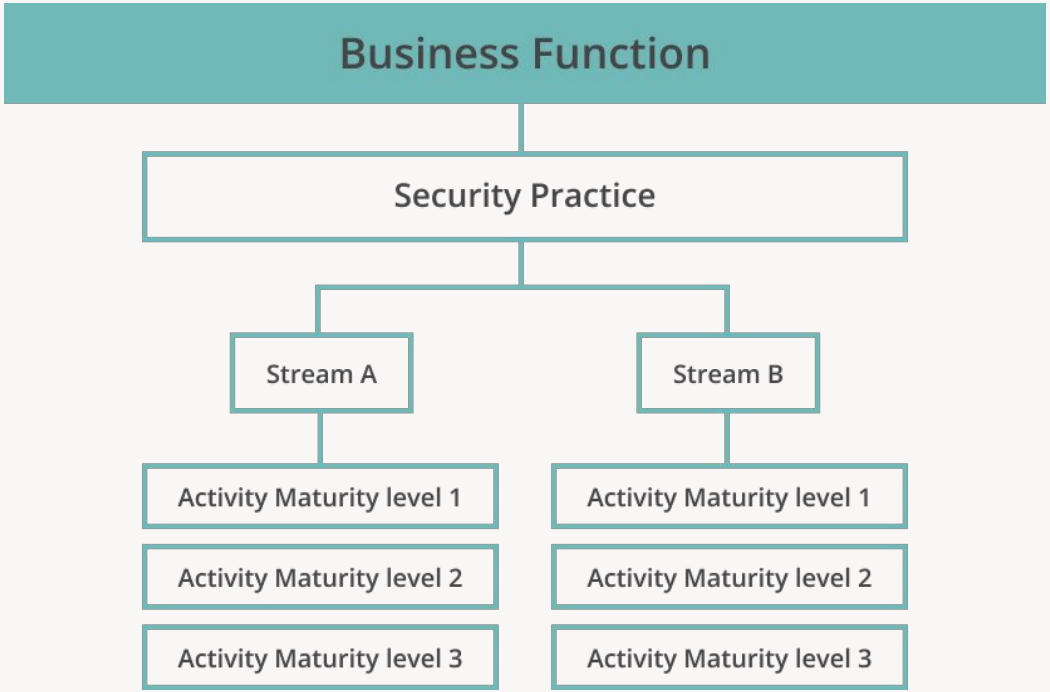
John Ellingsworth
Brian Glas

SAMM structure

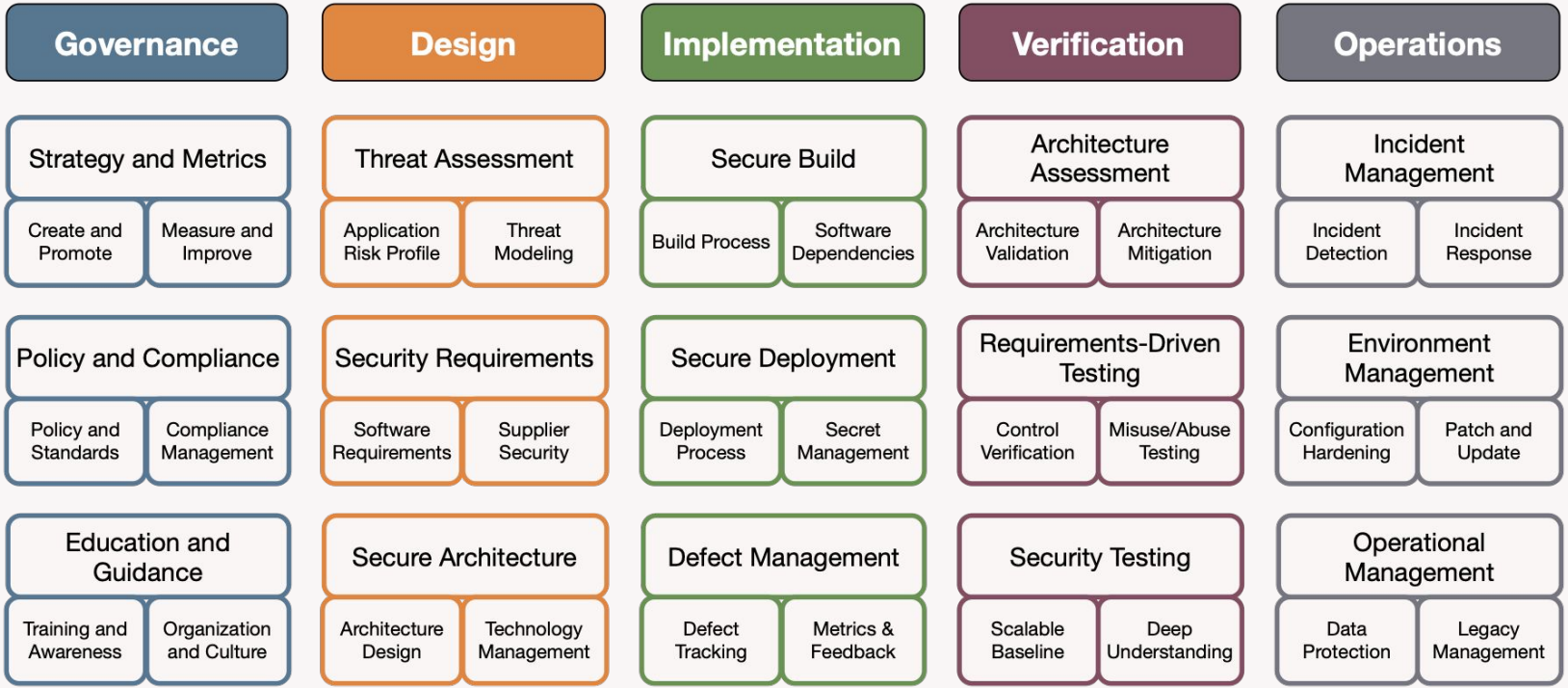
and how to run assessments

OWASP SAMM

SAMM v2 model structure



SAMM v2 Model overview



SAMM Activities (Education)

Maturity level	Training and awareness
Level 1: Ad-hoc provisioning	Provide security awareness training for all personnel involved in software development.
Level 2: Effectiveness and efficiency	Offer technology and role-specific guidance.
Level 3: Comprehensive mastery	Standardized in-house guidance around the organization's secure software development standards.

Training and awareness: level 1

Do you require employees involved with application development to take SDLC training?

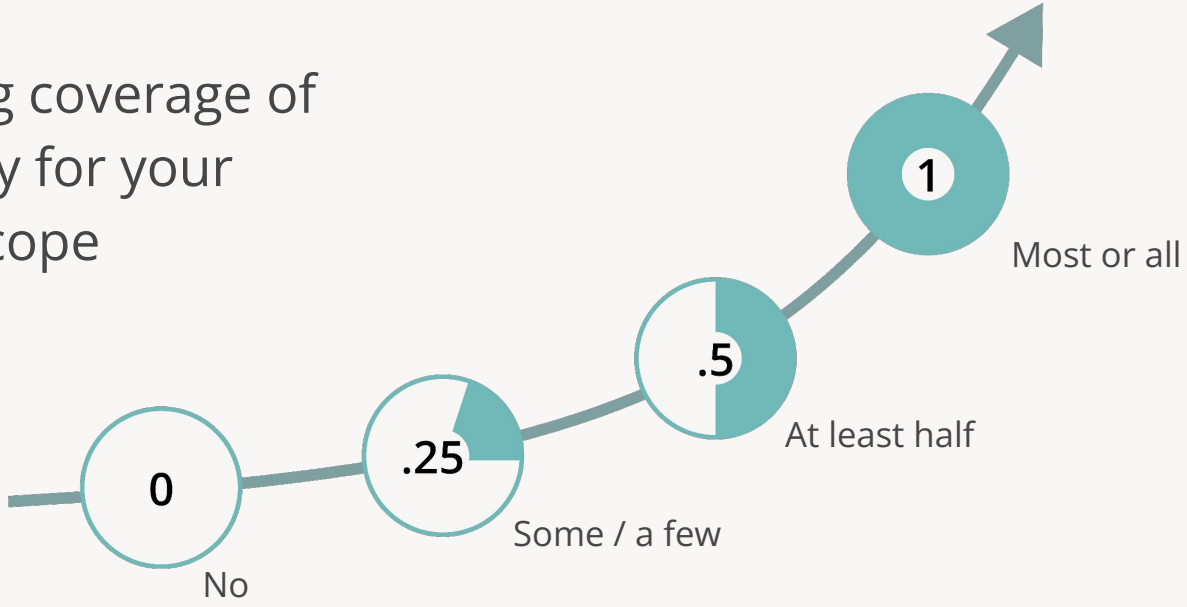
Answers
• No
• Yes, some of them
• Yes, at least half of them
• Yes, most of them

Quality criteria

- Training is repeatable, consistent, and available to anyone involved with SDLC
- Content updates in the last 12 months
- Training includes relevant content from the latest OWASP Top 10
- Training is required during onboarding

Assessment - scores

Measuring coverage of the activity for your defined scope



SAMM Activities (Secure Build)

Maturity level	Build process
Level 1: Ad-hoc provisioning	Create a formal definition of the build process so that it becomes consistent and repeatable.
Level 2: Effectiveness and efficiency	Automate your build pipeline and secure the used tooling. Add security checks in the build pipeline.
Level 3: Comprehensive mastery	Define mandatory security checks in the build process and ensure that building non-compliant artifacts fails.

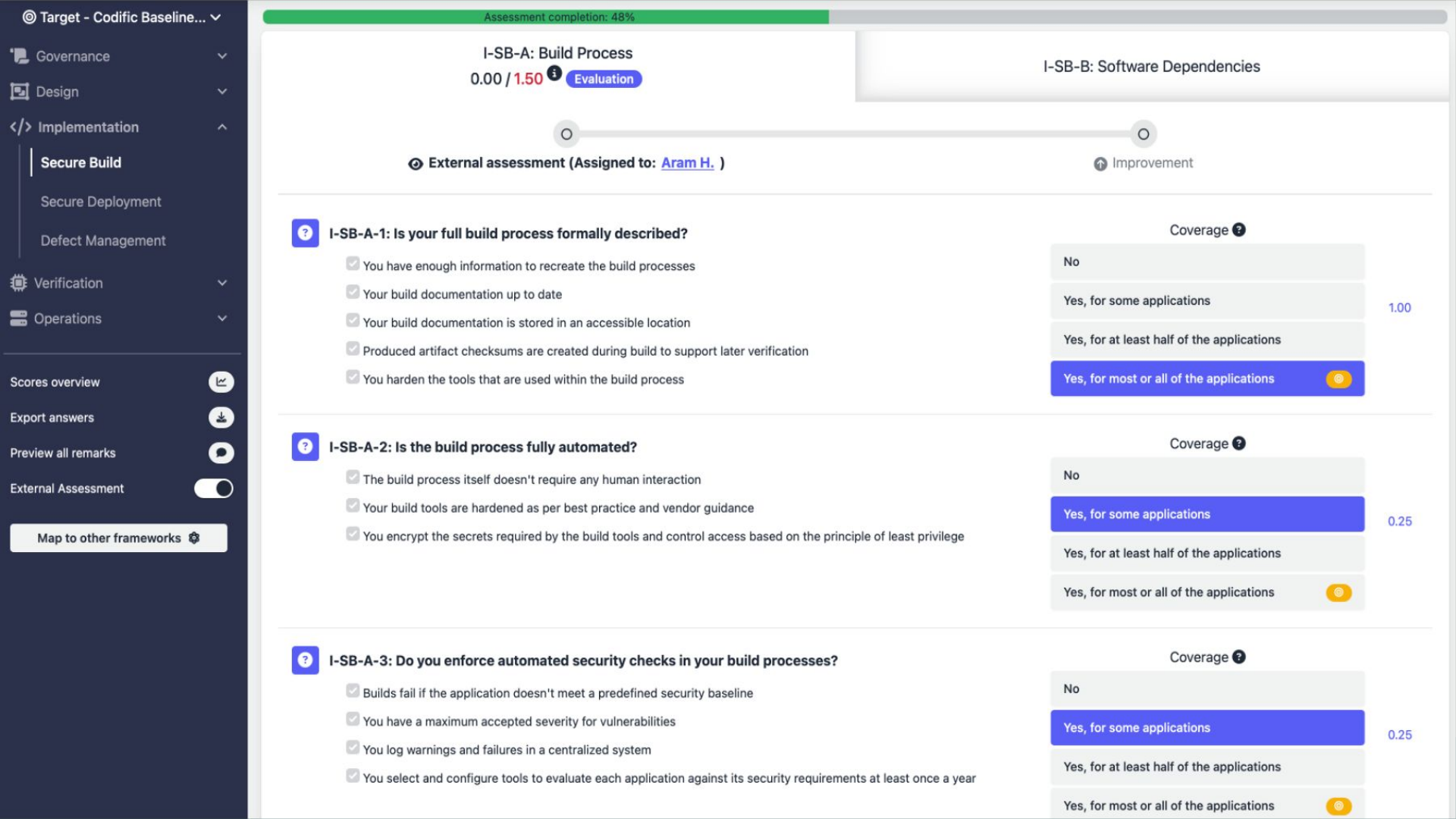
Build process: level 1

Is your full build process formally described?

Answers
• No
• Yes, for some apps
• Yes, for at least half of the apps
• Yes, for most or all of the apps

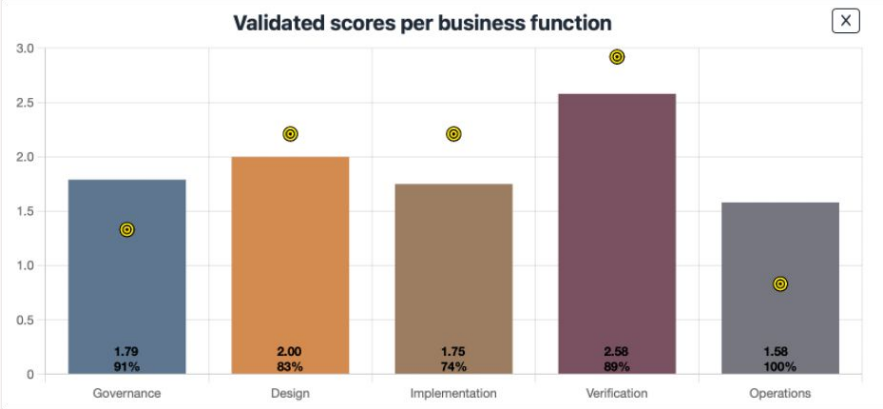
Quality criteria

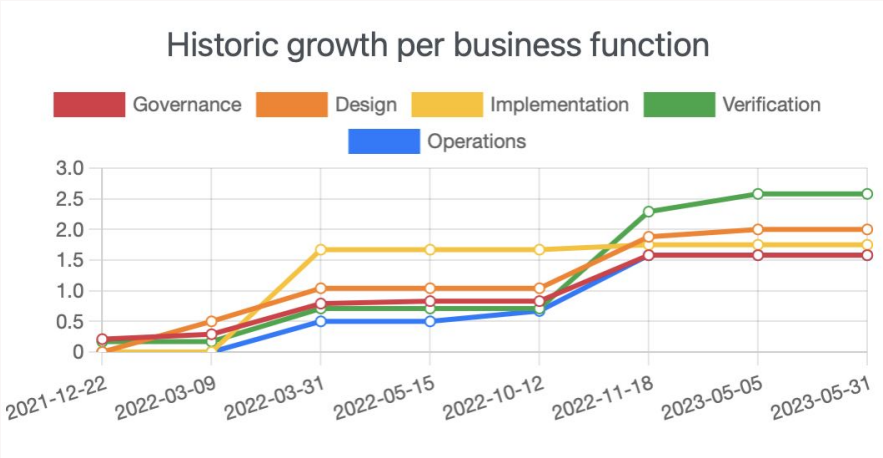
- Build documentation is up to date
- Build documentation is accessible
- Build creates artifact checksums
- You harden the build tools


Target - Codific Baseline...
Governance
Design
Implementation
Secure Build
Secure Deployment
Defect Management
Verification
Operations
Scores overview
Export answers
Preview all remarks
External Assessment
Map to other frameworks
Assessment completion: 48%
I-SB-A: Build Process
0.00 / 1.50
Evaluation
I-SB-B: Software Dependencies
External assessment (Assigned to: Aram H.)
Improvement
I-SB-A-1: Is your full build process formally described?
You have enough information to recreate the build processes
Your build documentation up to date
Your build documentation is stored in an accessible location
Produced artifact checksums are created during build to support later verification
You harden the tools that are used within the build process
Coverage
No
Yes, for some applications
Yes, for at least half of the applications
Yes, for most or all of the applications
1.00
I-SB-A-2: Is the build process fully automated?
The build process itself doesn't require any human interaction
Your build tools are hardened as per best practice and vendor guidance
You encrypt the secrets required by the build tools and control access based on the principle of least privilege
Coverage
No
Yes, for some applications
Yes, for at least half of the applications
Yes, for most or all of the applications
0.25
I-SB-A-3: Do you enforce automated security checks in your build processes?
Builds fail if the application doesn't meet a predefined security baseline
You have a maximum accepted severity for vulnerabilities
You log warnings and failures in a centralized system
You select and configure tools to evaluate each application against its security requirements at least once a year
Coverage
No
Yes, for some applications
Yes, for at least half of the applications
Yes, for most or all of the applications
0.25

Scorecards

- Evaluate existing practices
- Gap analysis
- Build a balanced improvement program
- Demonstrate improvements
- M&A assessments



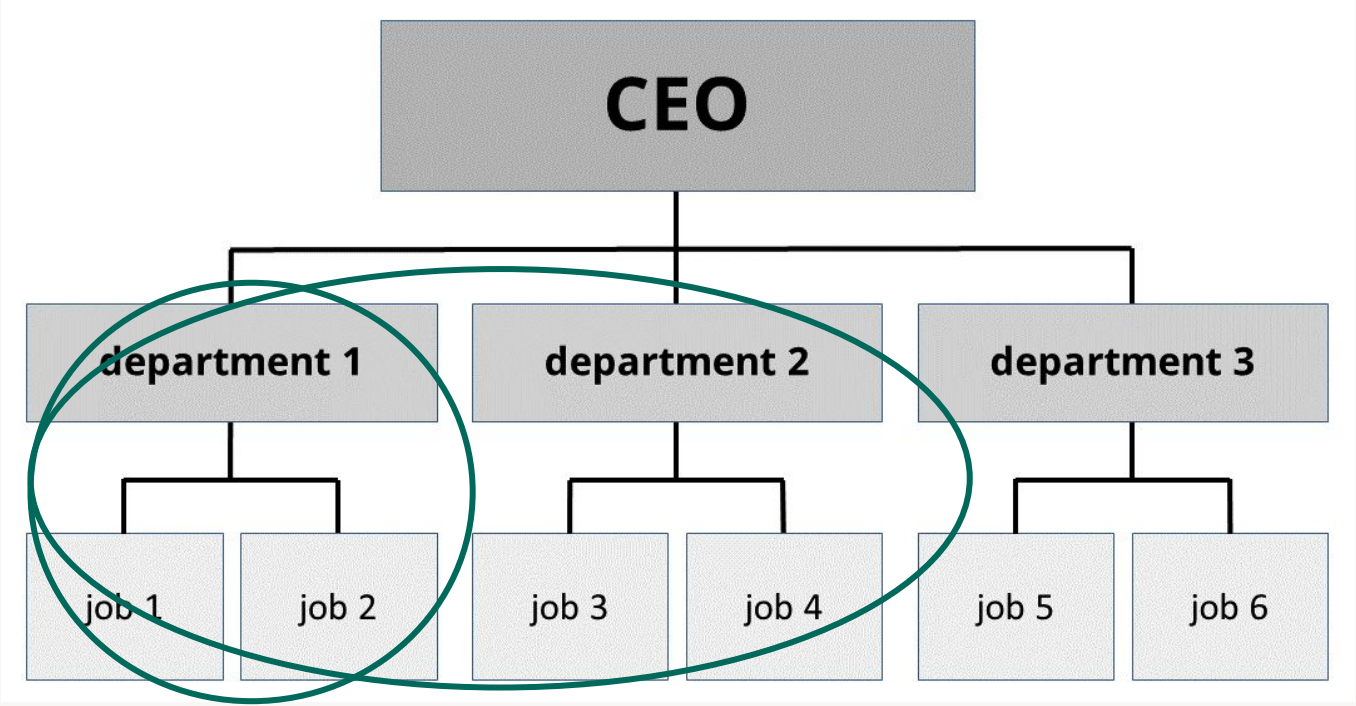


Methodology

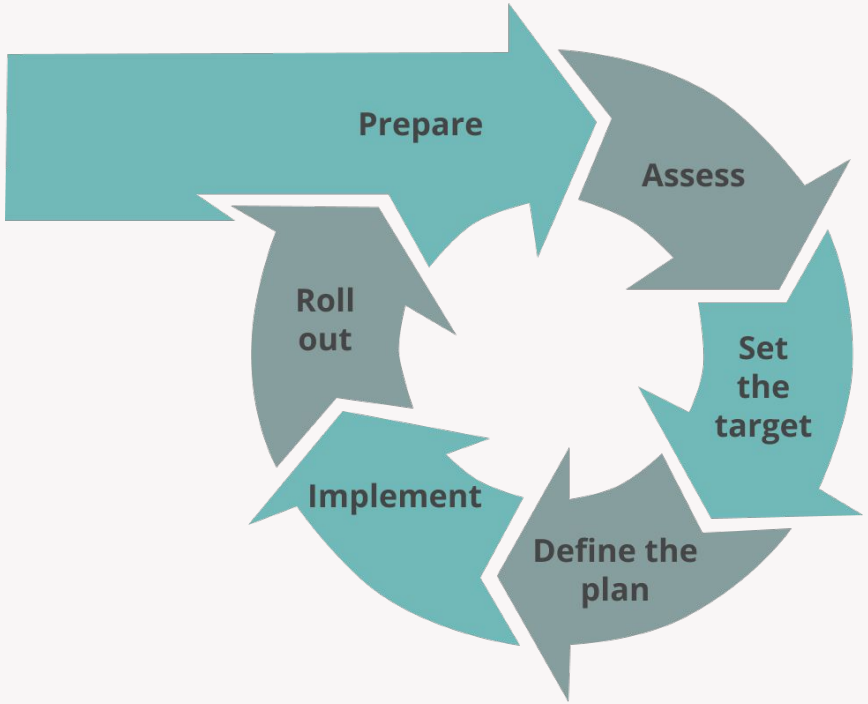
For using the model

OWASP SAMM

Set your scope



Methodology - steps

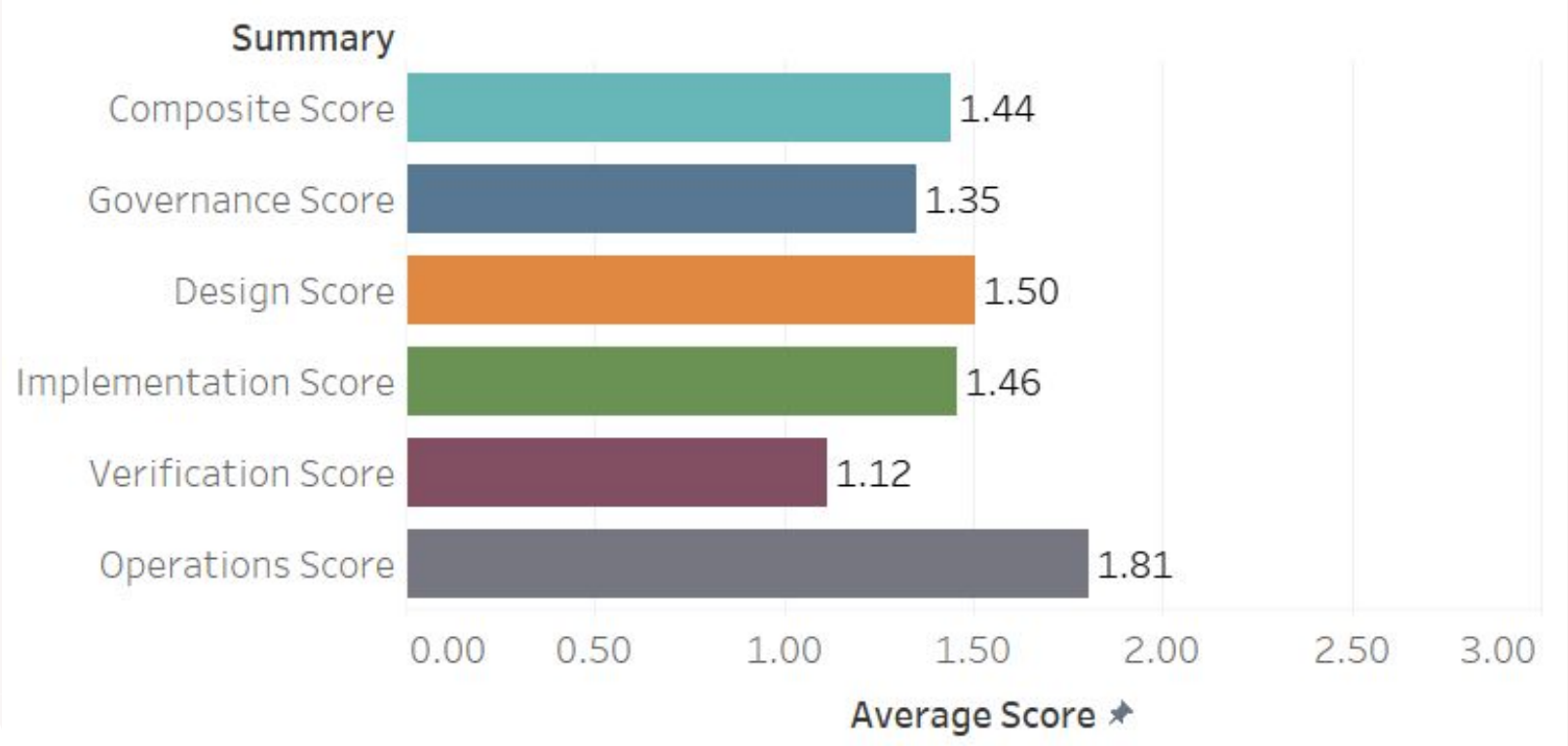


SAMM Benchmark

Benchmark tab in Excel Toolbox and SAMMY tool

https://bit.ly/sammbenchmarksubmission

Overall Results: 30 datasets



Secure Build

Demo

OWASP SAMM

Secure Build

Stream A: Build Process

Stream B: Software Dependencies

Build Process

L1: Is your full build process formally described?

- *You have enough information to recreate the build processes*
- *Your build documentation up to date*
- *Your build documentation is stored in an accessible location*
- *Produced artifact checksums are created during build to support later verification*
- *You harden the tools that are used within the build process*

- No
- Yes, for some applications
- Yes, for at least half of the applications
- **Yes, for most or all of the applications**

Build Process

L2: Is the build process fully automated?

- *The build process itself doesn't require any human interaction*
- *Your build tools are hardened as per best practice and vendor guidance*
- *You encrypt the secrets required by the build tools and control access based on the principle of least privilege*

- No
- Yes, for some applications
- Yes, for at least half of the applications
- **Yes, for most or all of the applications**

Build Process

L3: Do you enforce automated security checks in your build processes?

- *Builds fail if the application doesn't meet a predefined security baseline*
- *You have a maximum accepted severity for vulnerabilties*
- *You log warnings and failures in a centralized system*
- *You select and configure tools to evaluate each application against its security requirements at least once a year*

- No
- Yes, for some applications
- Yes, for at least half of the applications
- **Yes, for most or all of the applications**

Software Dependencies

L1: Do you have solid knowledge about dependencies you're relying on?

- *You have a current bill of materials (BOM) for every application*
- *You can quickly find out which applications are affected by a particular CVE*
- *You have analyzed, addressed, and documented findings from dependencies at least once in the last three months*

- No
- Yes, for some applications
- **Yes, for at least half of the applications**
- Yes, for most or all of the applications

Software Dependencies

L2: Do you handle 3rd party dependency risk by a formal process?

- *You keep a list of approved dependencies that meet predefined criteria*
- *You automatically evaluate dependencies for new CVEs and alert responsible staff*
- *You automatically detect and alert to license changes with possible impact on legal application usage*
- *You track and alert to usage of unmaintained dependencies*
- *You reliably detect and remove unnecessary dependencies from the software*

No

Yes, for some applications

Yes, for at least half of the applications

Yes, for most or all of the applications

Software Dependencies

L3: Do you prevent build of software if it's affected by vulnerabilities in dependencies?

- *Your build system is connected to a system for tracking 3rd party dependency risk, causing build to fail unless the vulnerability is evaluated to be a false positive or the risk is explicitly accepted*
- *You scan your dependencies using a static analysis tool*
- *You report findings back to dependency authors using an established responsible disclosure process*
- *Using a new dependency not evaluated for security risks causes the build to fail*

No

Yes, for some applications

Yes, for at least half of the applications

Yes, for most or all of the applications

Wrap-up

OWASP SAMM

Conclusion

- Application Security is a challenging problem
  - Complex
  - Broad
  - Evolving
- AppSec requires a continuous assurance programme
- SAMM is a simple, well-defined and measurable maturity model

Thank you!

https://www.linkedin.com/in/aramhovsep

https://www.linkedin.com/company/codific