SECURING SOFTWARE AT SCALE

A MODERN SECURE DEVELOPMENT APPROACH

Michael Helwig

🏠 Codemetrix GmbH

@c0dmtr1x

codemetrix
SECURITY | SOFTWARE | DATA PROTECTION

APPLICATION ATTACKS ARE A GROWING BUSINESS RISK

Business Case

- Business-Continuity
- GDPR & Compliance
- Reputation

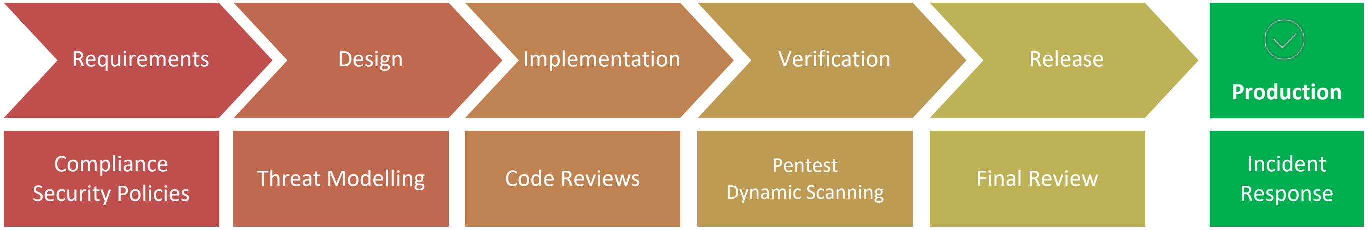
Applications provide a common attack surface

- Applications play a growing role for businesses
- Applications keep growing as attack plane[1]
- Responsible for 41% of data breaches[2]
- Company size does not protect you[1]

Common Application Threats [1,3]

- Injection attacks
- Open Source Vulnerabilities
- Broken Authentication
- Sensitive Data exposure
- Etc.

Solution: SSDLC



[1] Forrester: *State of Application Security 2018*
[2] Verizon: *DBIR 2018*
[3] OWASP: *TOP10:2018*

…BUT SOFTWARE GETS DELIVERED FASTER AND OLD SOLUTIONS ARE TOO SLOW

Before	Now
Waterfall Approach	Agile
Requirements & Design phases	User Stories
Slow Release Cycles	Sprints
Manual Testing & Deployment Pipelines	CI / CD Pipelines
Monolithic	Microservices
Separation Between Dev & Ops	DevOps

Security	Problems
Security Profile	No clear project specification
Threat Modelling	“No time for that!”
Pentests before Release	“Takes at least 1 week plus fixing time”
Final Review	“You should have told us before”
Cover whole application from E2E	“We don‘t know where this data goes. Is it prod?”
Just deploy, scan for vulnerabilities in Production	“We coded this? Cannot remember!”

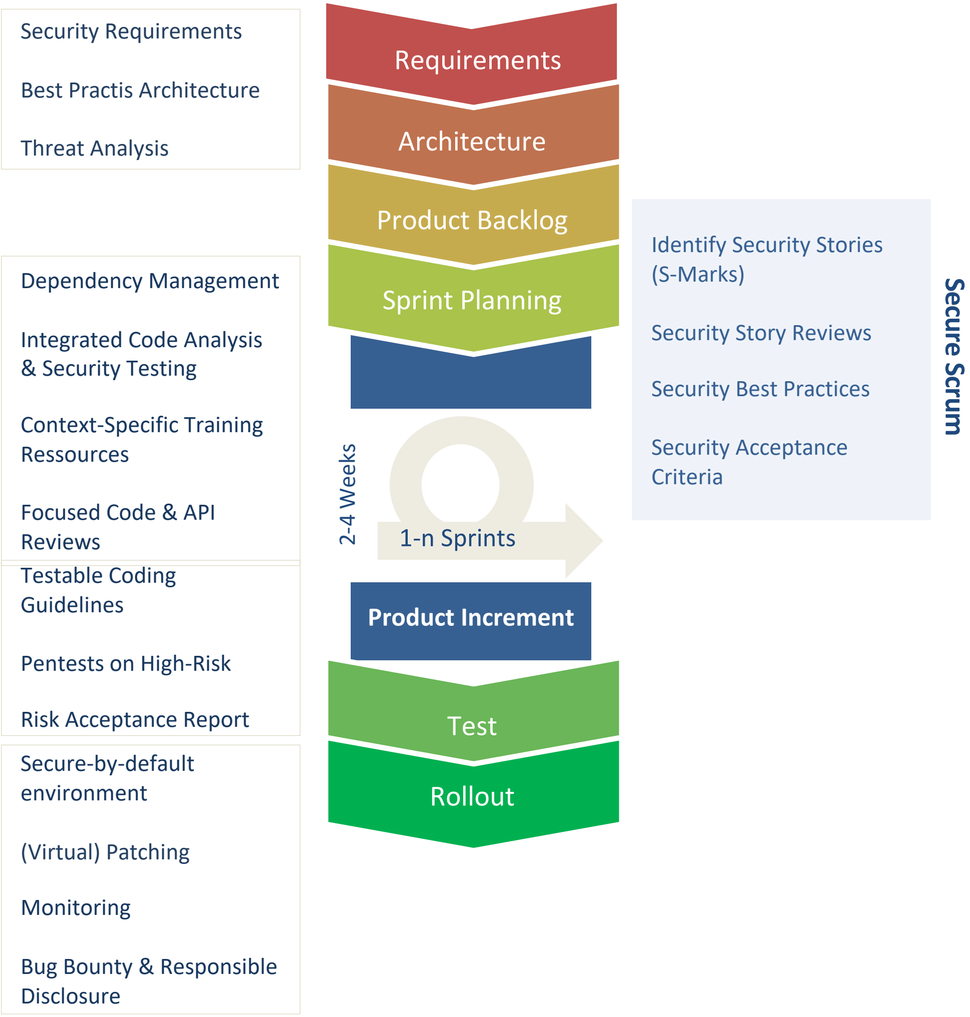
TODAY'S SECURITY NEEDS TO WORK FAST AND ADAPT

Shift-Left

- Provide Developers feedback as soon as possible
- Use AST tools with IDE Integration and low FP rate
- Open source analysis
- Default solutions and security patterns
- Integrate Security into Scrum (Secure Scrum, Abuse Stories)
- Developer security training

DevSecOps

- Integrate security tests into the pipeline: SAST, IAST, DAST
- „Test early and often“
- Reviews focused on critical code changes, e.g. on authentication
- Secure deployment enironments, e.g. an internal container registry with scanned images



Security Requirements

Best Practis Architecture

Threat Analysis

Dependency Management

Integrated Code Analysis & Security Testing

Context-Specific Training Ressources

Focused Code & API Reviews

Testable Coding Guidelines

Pentests on High-Risk

Risk Acceptance Report

Secure-by-default environment

(Virtual) Patching

Monitoring

Bug Bounty & Responsible Disclosure

Secure Scrum

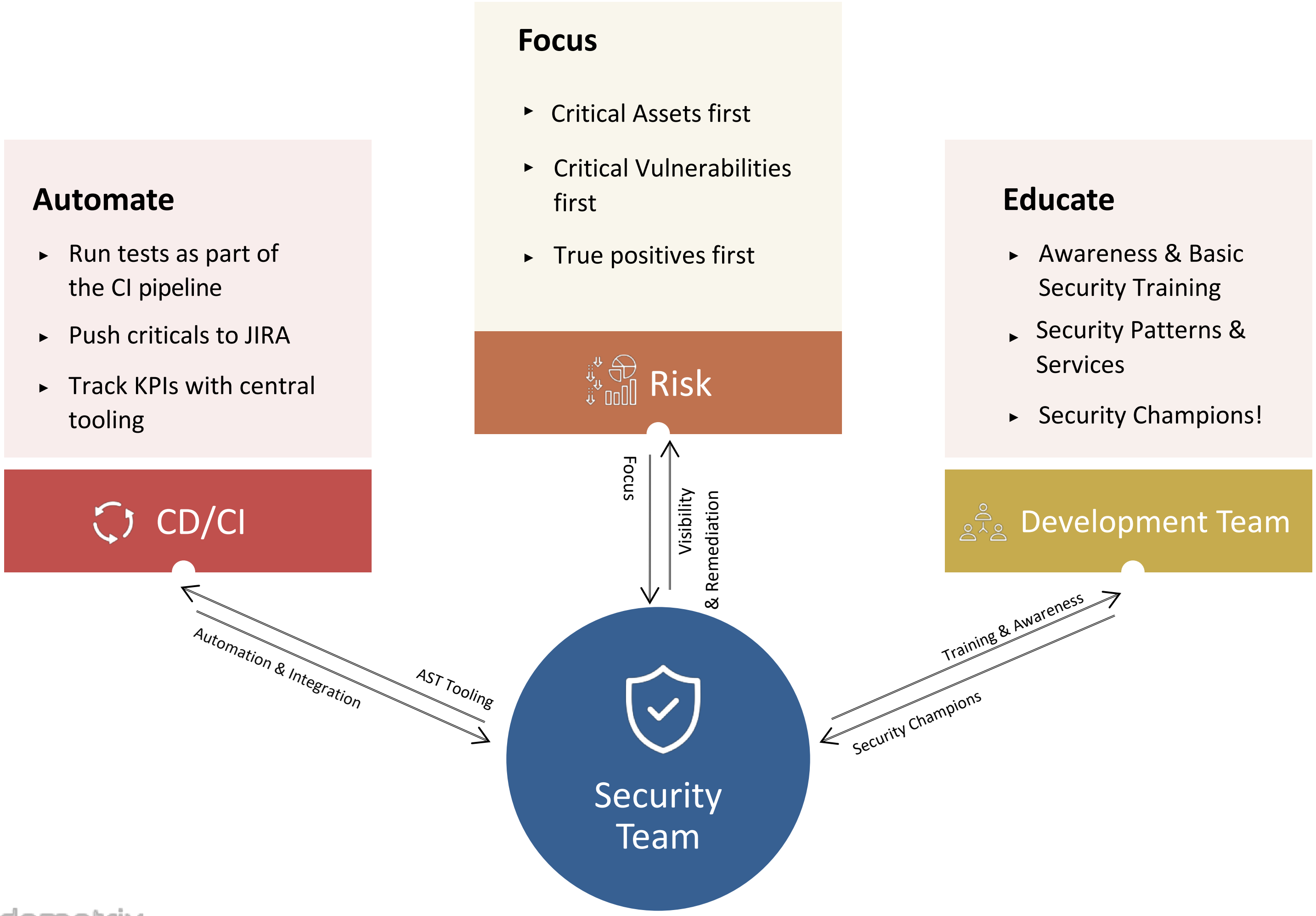
Identify Security Stories (S-Marks)

Security Story Reviews

Security Best Practices

Security Acceptance Criteria

… AND SCALE IN A LARGE COMPANY WITH A SMALL TEAM!



Automate

- Run tests as part of the CI pipeline
- Push criticals to JIRA
- Track KPIs with central tooling

CD/CI

Focus

- Critical Assets first
- Critical Vulnerabilities first
- True positives first

Risk

Educate

- Awareness & Basic Security Training
- Security Patterns & Services
- Security Champions!

Development Team

Security Team