Mastering Security in Agile/Scrum, Case Study

Anu Puhakainen & Juha Sääskilahti

Ericsson

Network Security Competence Hub

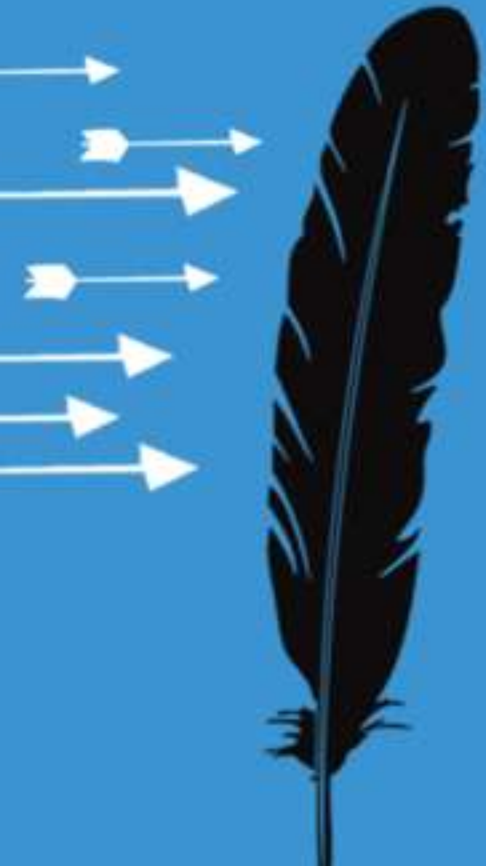
Session ID: ASEC-107

Session Classification: Intermediate

RSACONFERENCE EUROPE 2012

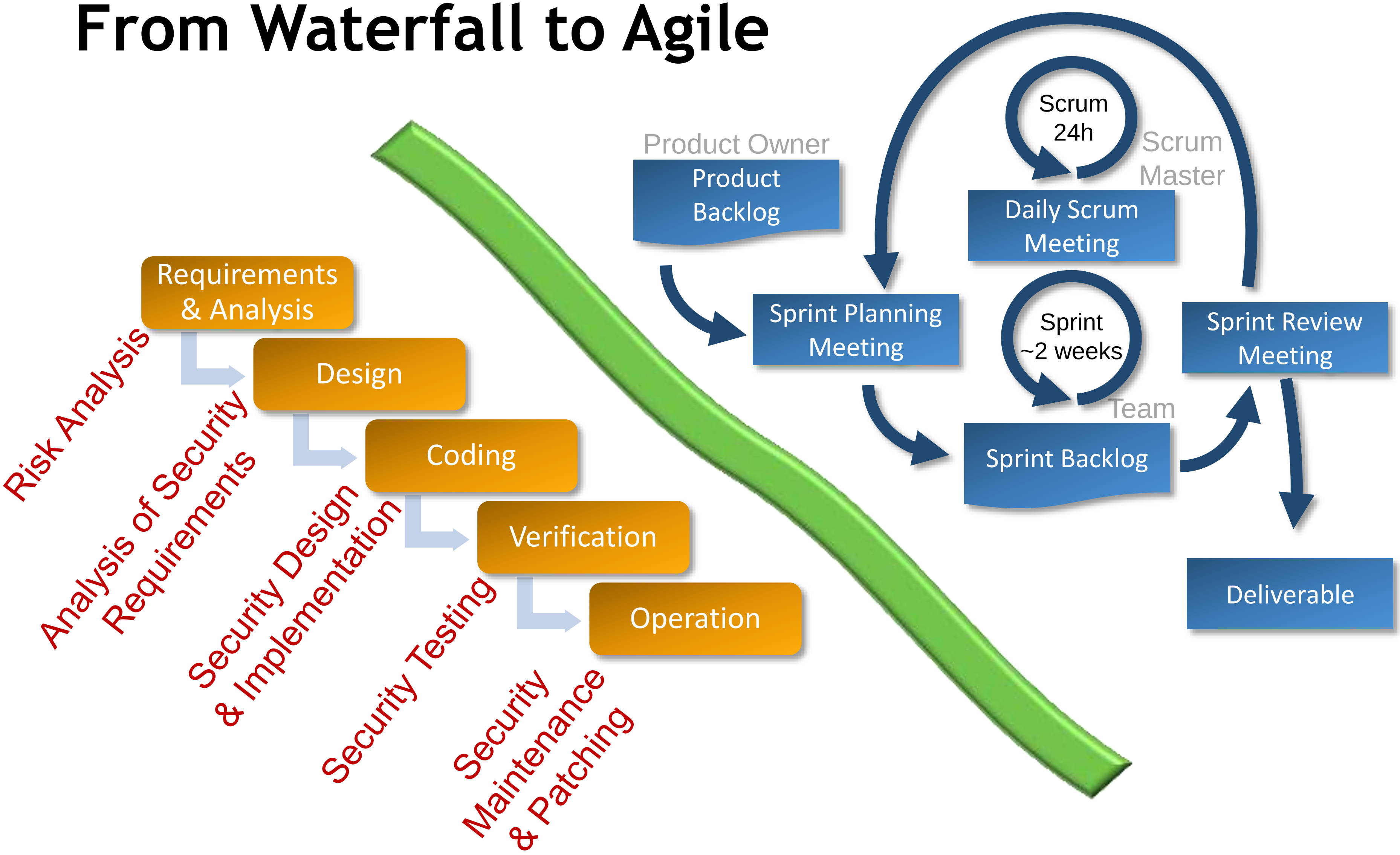
Presentation Outline

- Introduction
  - Agile Transformation background
- Case Study: Security in Agile
- Five takeaways



Introduction

From Waterfall to Agile



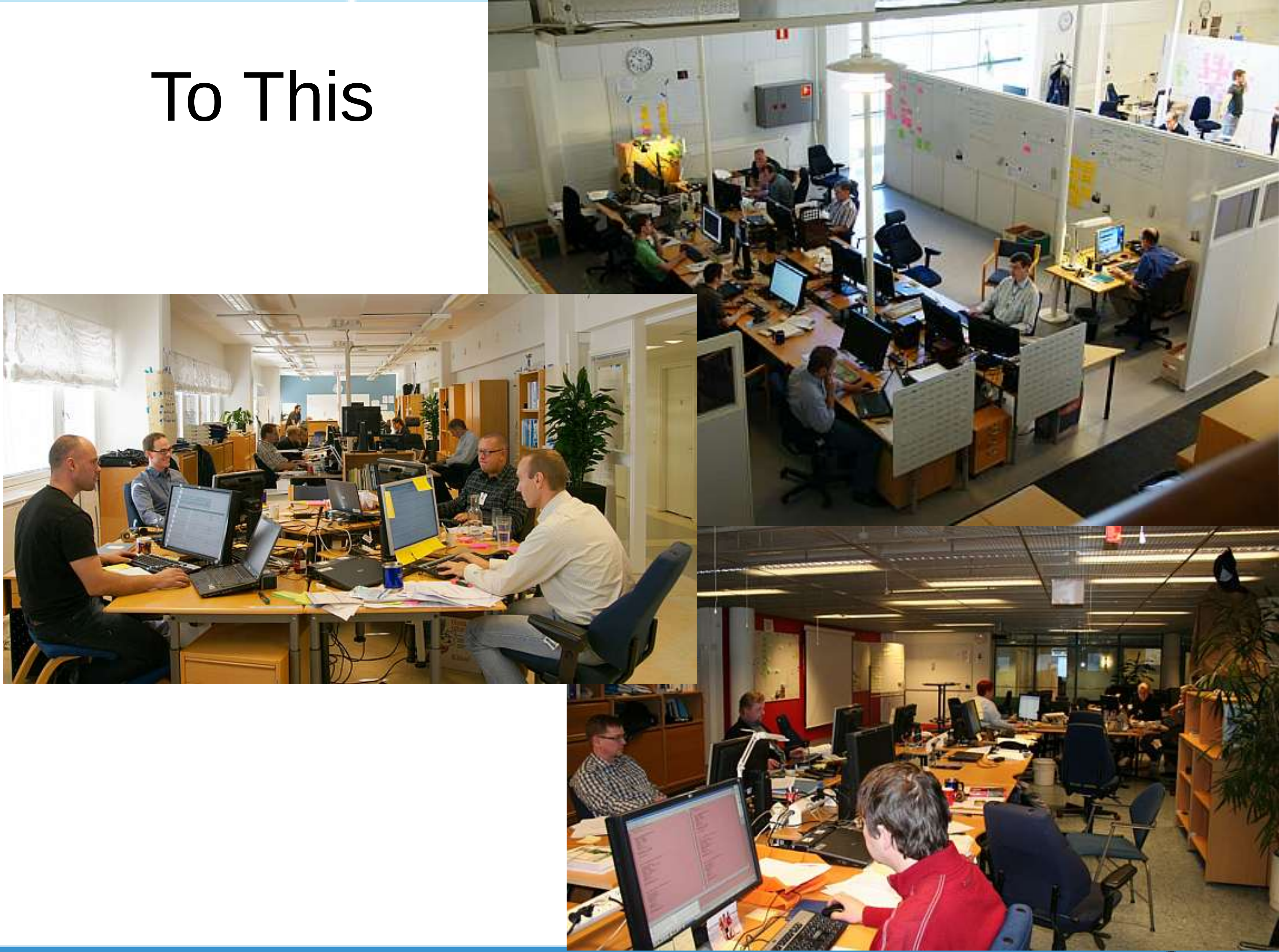
Agile Transformation

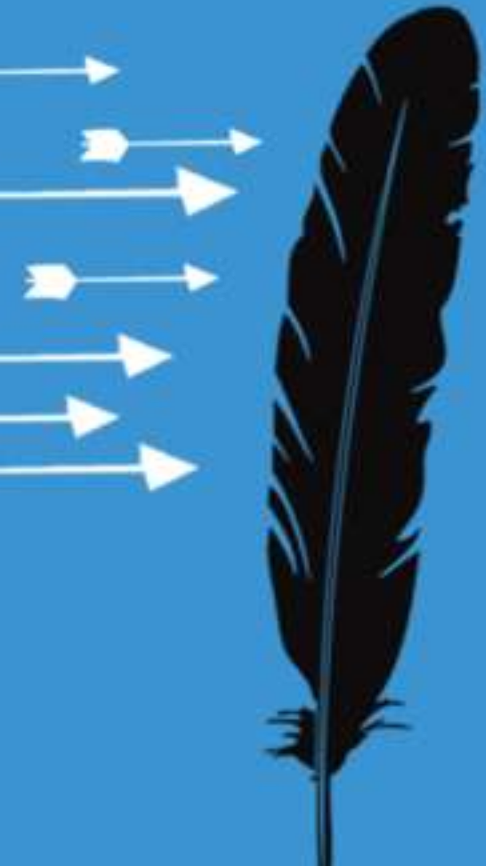
- Major R&D Agile Transformation
  - Ericsson Finland as forerunner ~500 R&D employees working in software development for mobile networks
- Not only process change – also a big cultural change!

From This

Through This

To This





CASE STUDY: Security in Agile

Background to the Case Study

Research partners with major interest in agile security

- R&D Transformation Case linked to
  - TiViT Cloud SW Research Project initiated 2010
    - Multi-branched research, including Agile
- Problem statement for Security in Agile
  - Current Agile/Scrum models do not have security embedded

What have we researched until now?

- Agile Transformation – yes
- But ... How is Security embedded?
  - How to make sure products developed with agile/scrum/lean are secure?
- Develop good practice for global Ericsson R&D
  - Theory meets practice – or does it?

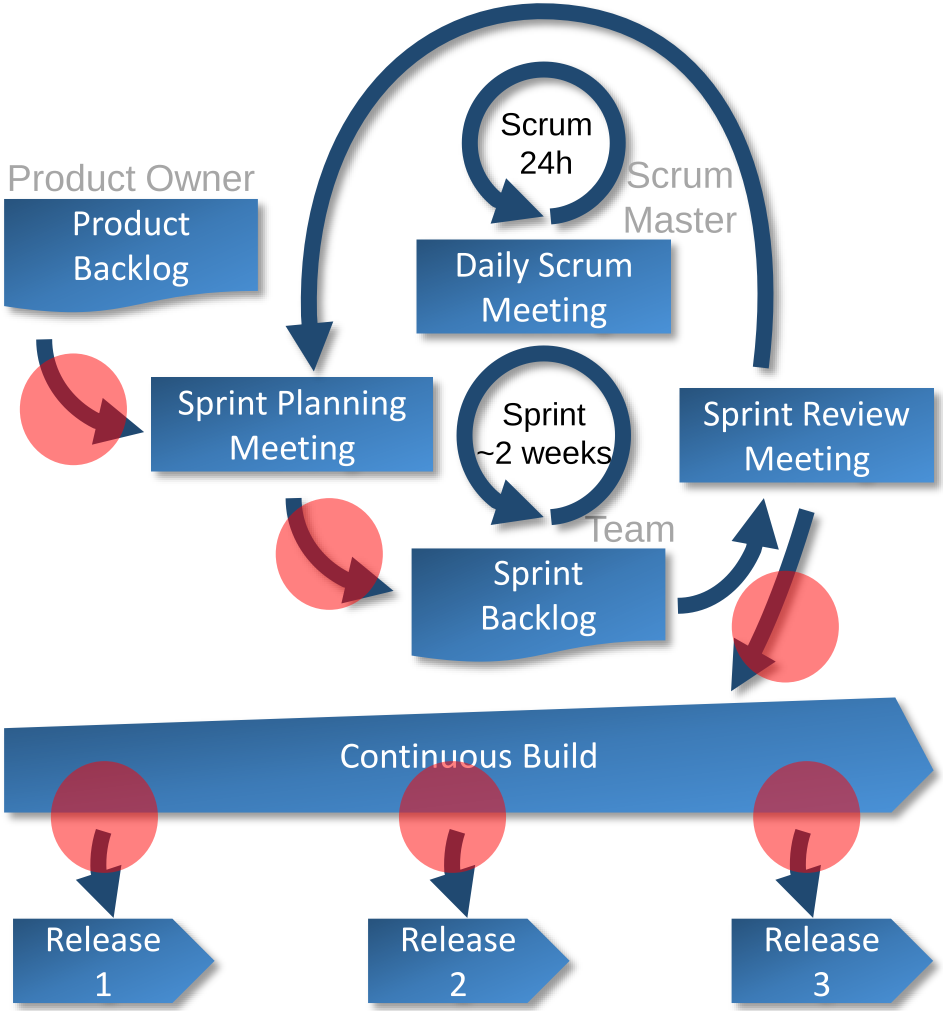
Starting Point - Risk Analysis (RA)

- Old methodology
  - Suited for product releases with relatively long interval
- Agile brings new requirements
  - More frequent product releases
  - More dynamic feature changes (short lead time)

Tangible outcome: New RA method

- Promises:
  - Minimal preparation work required prior to workshop
  - Workshop of ½ - 2 days for a full product
    - For new features, very quickly … 15min(?)
  - More fluid workshops; mind-maps instead of matrixes
    - More motivating for participants
    - Using xMind (but any mind-map is ok)
  - Templates
- Iterated and experimented 10-15 times before outlining Agile RA methodology

Risk Management with Agile/ Continuous Integration



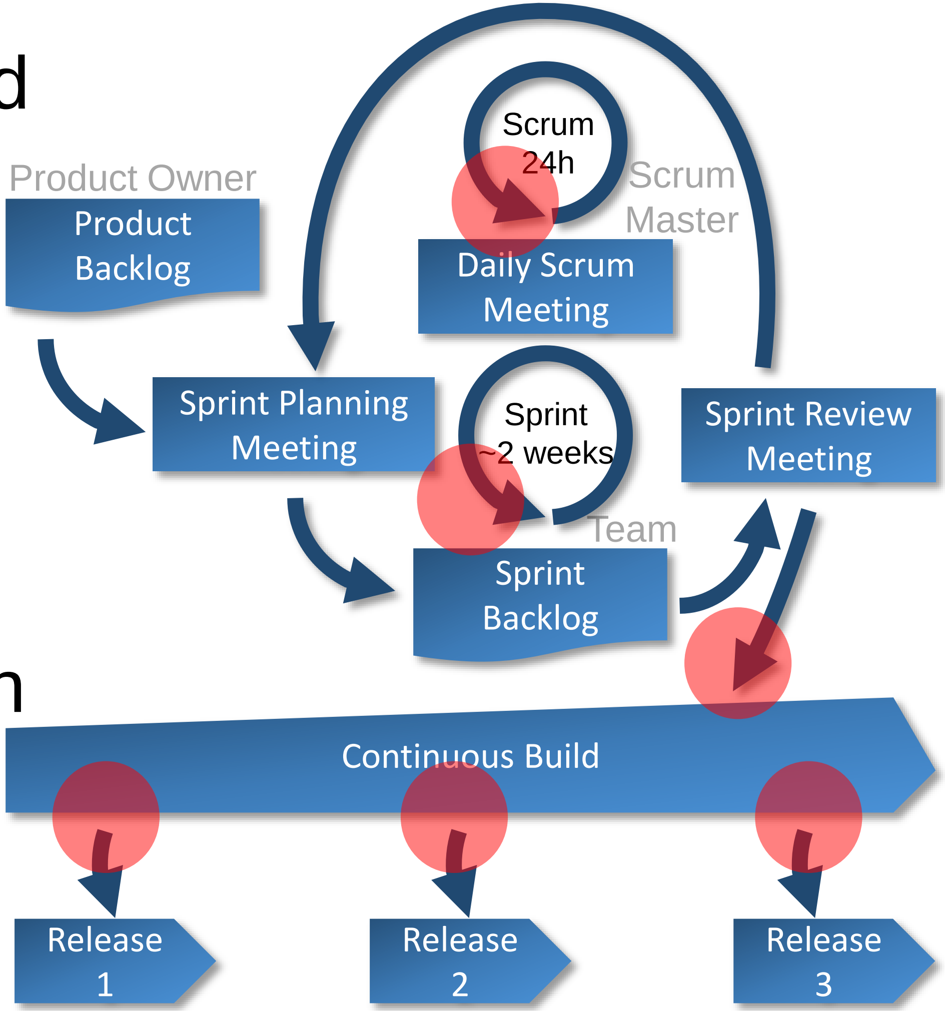
- Business Level Risk Analysis – updated every time product backlog changes
- Technical Level Risk Analysis – every time sprint starts
- Validation of Risks
  - Every check-in
  - Every Product Release

What else has been achieved so far?

- Security awareness – one key learning
  - Security much more visible now
- Learning from other companies and organizations
  - Research consortium, SafeCode…
  - Don’t try this alone at home!

Next area to address in detail: Security Testing

- Objective to find a good working model:
  - Set 1: Every Check-In
  - Set 2: Every Scrum
  - Set 3: Every Epic/sprint
  - Set 4: Every Product Release to Customer
- Automate what you can
  - Some tests should not too be automated

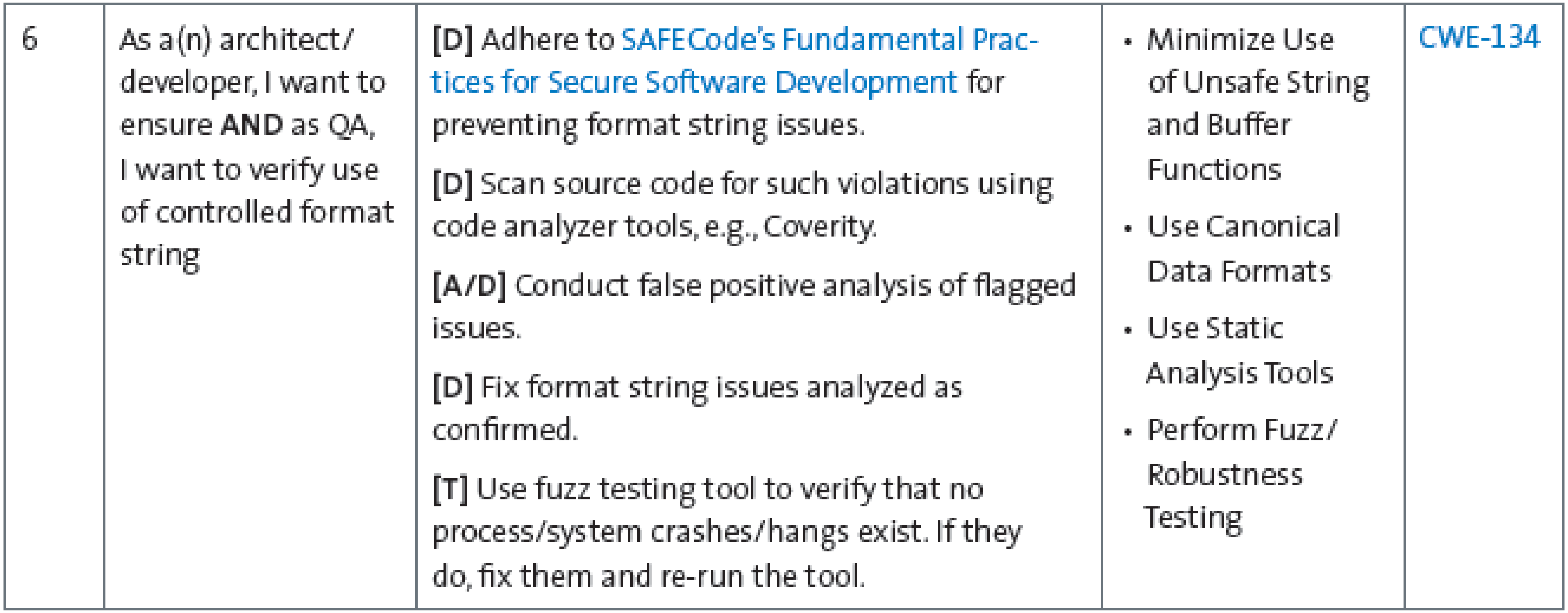


Security Requirements Management in Agile

- Non-functional requirements (e.g. Security requirements) – challenge in Agile
- 2 fundamental problems
    - Which requirements to choose
    - How to formulate the chosen requirements into Agile User Stories
        - Negative user stories? – How to confirm by testing?

Example

SAFECode
Software Assurance Forum for Excellence in Code
Driving Security and Integrity

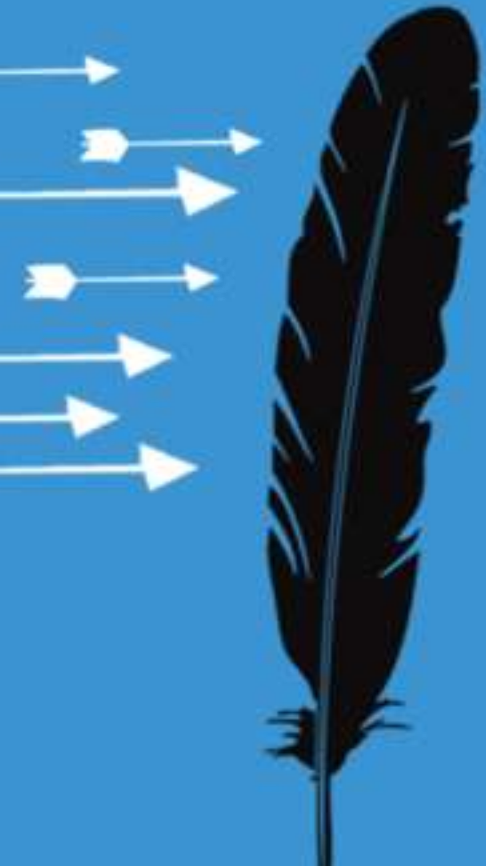


6	As a(n) architect/ developer, I want to ensure **AND** as QA, I want to verify use of controlled format string	[D] Adhere to SAFECode's Fundamental Practices for Secure Software Development for preventing format string issues. [D] Scan source code for such violations using code analyzer tools, e.g., Coverity. [A/D] Conduct false positive analysis of flagged issues. [D] Fix format string issues analyzed as confirmed. [T] Use fuzz testing tool to verify that no process/system crashes/hangs exist. If they do, fix them and re-run the tool.	• Minimize Use of Unsafe String and Buffer Functions • Use Canonical Data Formats • Use Static Analysis Tools • Perform Fuzz/ Robustness Testing	CWE-134

Software Security Guidance for Agile Practitioners
www.safecode.org

And we continue with these as well

- Processes
  - Finalizing Process – security control points
    - How to add controls without sacrificing ’agile model’
- Organization
  - Who should have which competence?
- Measuring security...
  - No good metrixes for product security



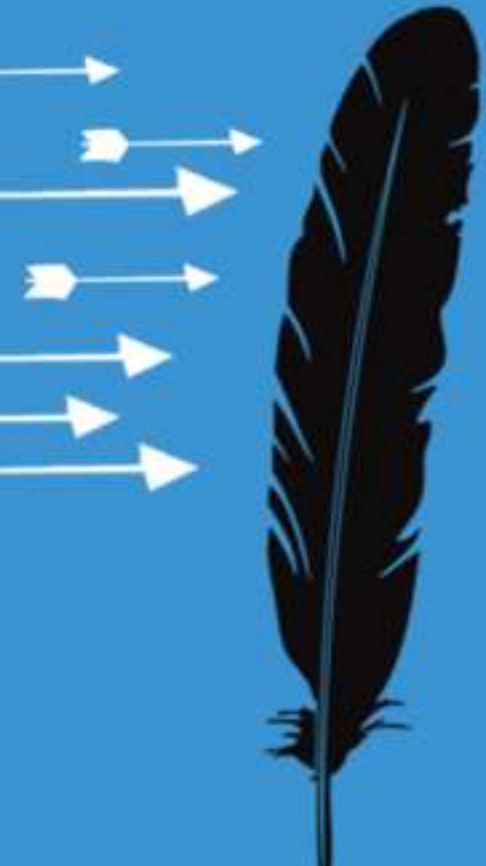
Takeaways

How to Apply Security in Agile

- Apply security ’agilely’
  - Bit by bit; no ’one-big-shot’
  - Adjust on the fly, give room for iteration
- Allocate sufficient resources
- Take learnings from other companies
- Make use of existing material

For security in agile, define strategy for:

- Organization – Security Roles, Responsibilities
- Process – Security Control Points
- Security Requirements Management
- Risk Management
- Security Verification



Questions?