AGIL, ABER SICHER?
SECURE SOFTWARE ENGINEERING

11.5.2016, ANDREAS FALK, UNI TÜBINGEN SOFTWARE ENGINEERING

NOVATEC | EBERHARD KARLS UNIVERSITÄT TÜBINGEN

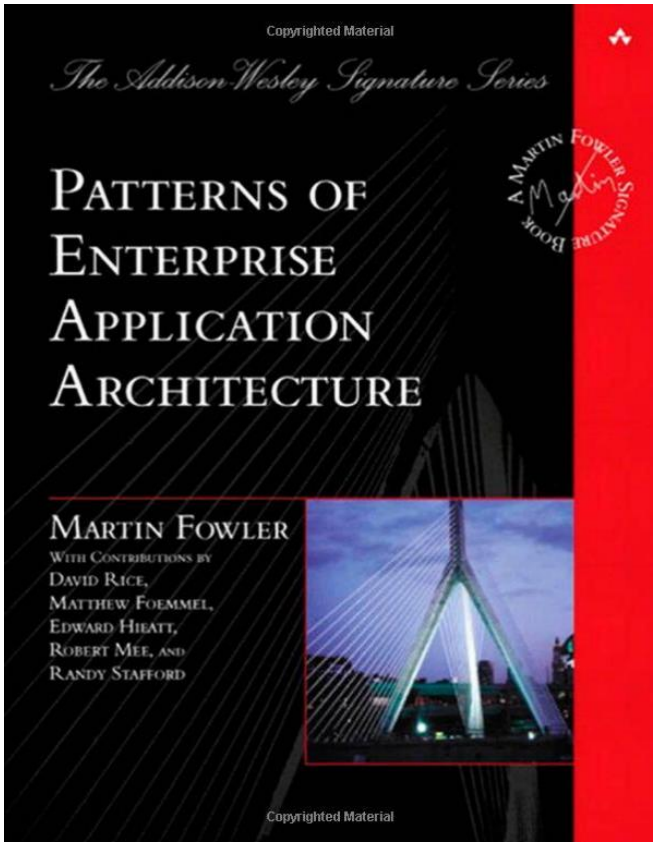
Über Mich

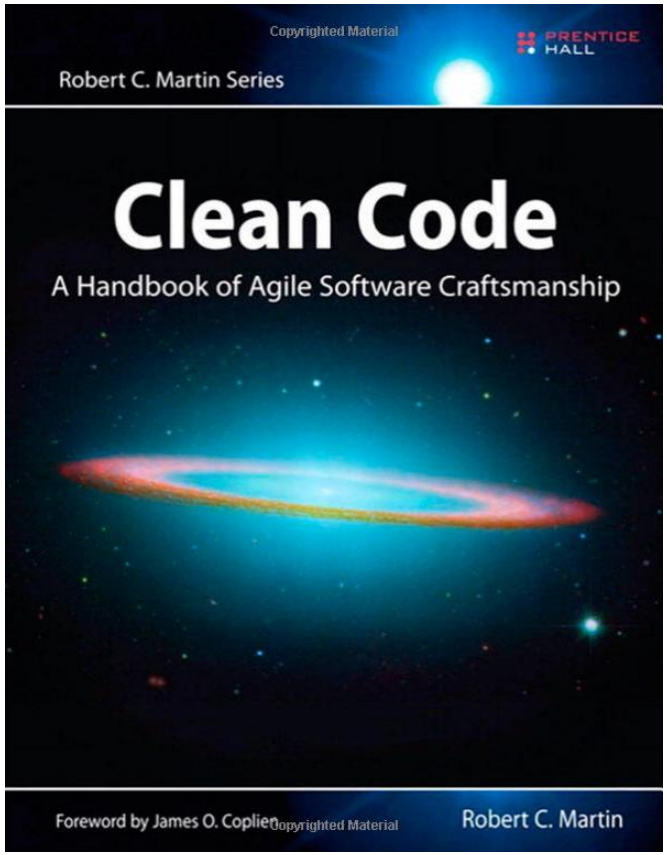
Andreas Falk
NovaTec Consulting GmbH
andreas.falk@novatec-gmbh.de

Mitglied der OWASP

@NT_AQE
@Agile_Security

Wichtige Literatur





Inhalte dieser Session

Wichtigkeit von Security bewusst machen!

Agilität und Security – Wie geht das zusammen?

Security-Grundlagen: Keine „Hacker“-Session!

https://creativecommons.org/licenses/by-sa/4.0/deed.en

Wirklich sicher? Hacks können Leben verändern!

Wirklich sicher? https://www.shodan.io

Troy Hunt @troyhunt · 9 Std.

Internet of Things security is so bad, there's a search engine for sleeping kids arstechnica.com/security/2016/...

Übersetzung anzeigen

"Internet of Things" security is hilariously broken and getting worse

Shodan search engine is only the latest reminder of why we need to fix IoT security.

Wirklich sicher? Verschlüsselung „a la“ stackoverflow.com

This is some what simple

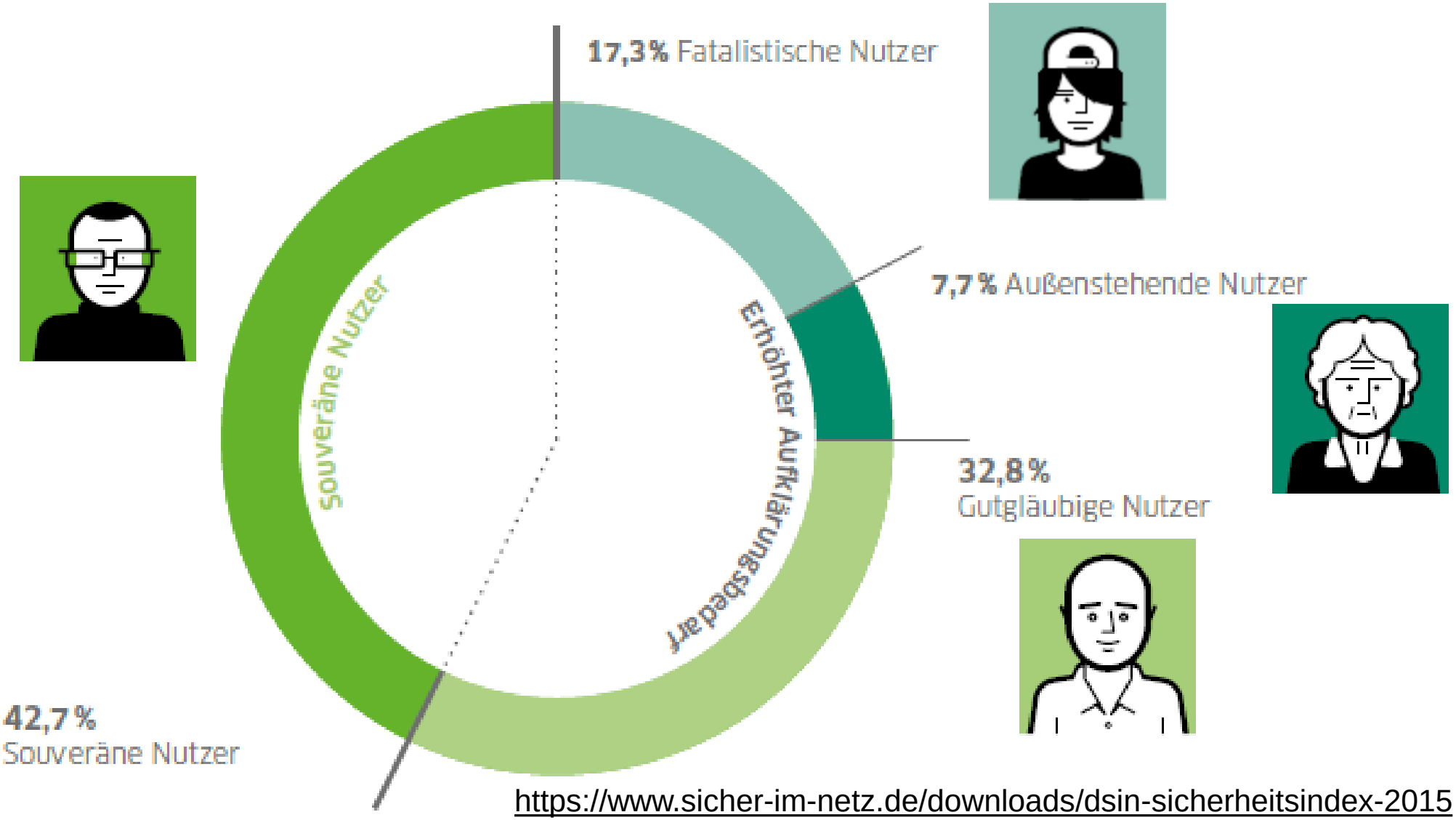
```
string inp = "hai";
StringBuilder strb = new StringBuilder();
foreach (char s in inp)
{
    int sin = s + 5;
    char newch = (char)sin;
    strb.Append(newch);
}
string output = strb.ToString();
```

Now the output contains the encrypted string "mfn" (ie., 5 letters away from the original)in it....

Verschiebechiffre:

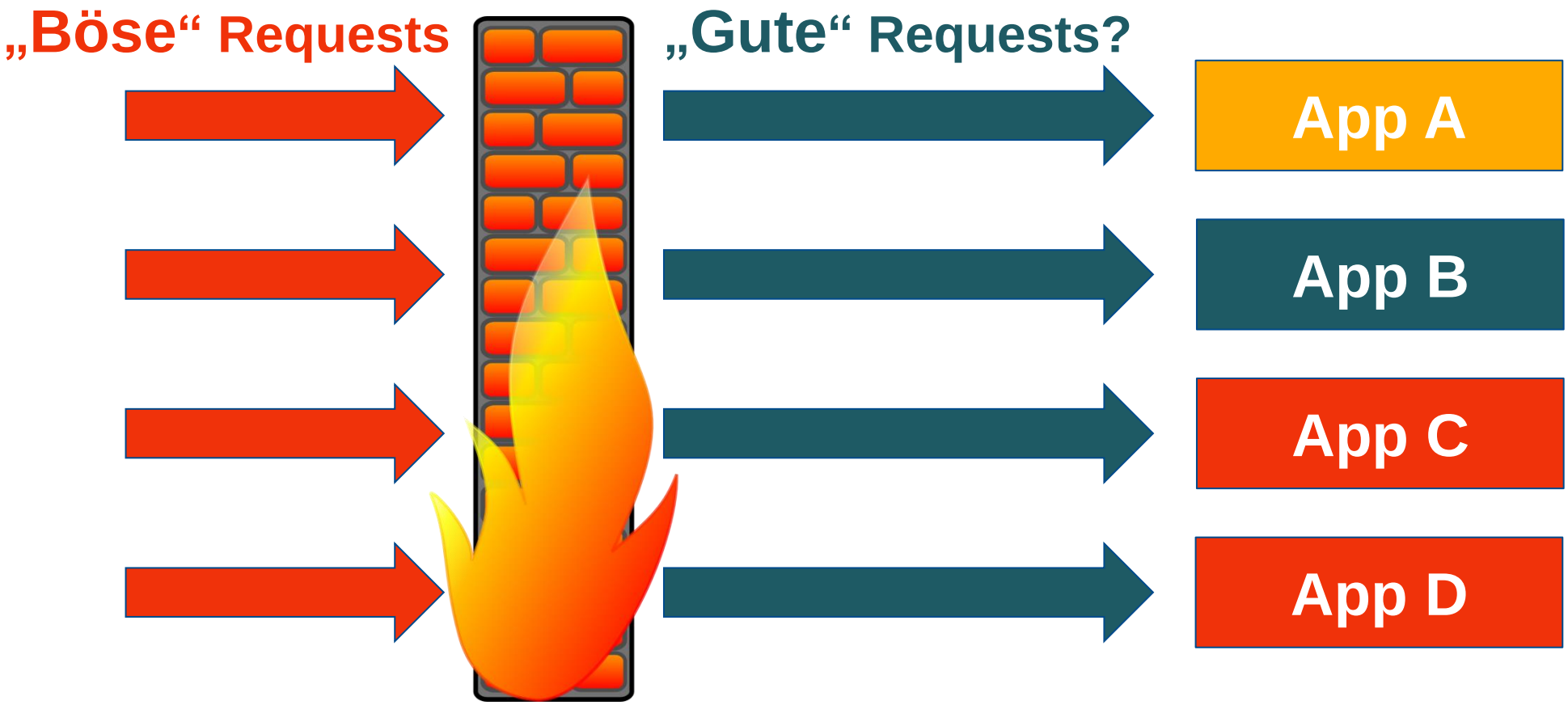
https://de.wikipedia.org/wiki/ROT13

Wirklich sicher? Nutzerverhalten!



https://www.sicher-im-netz.de/downloads/dsin-sicherheitsindex-2015

Wir haben doch eine „Security“-Firewall !?



„The Great Firewall“

Potentiell unsichere Systeme

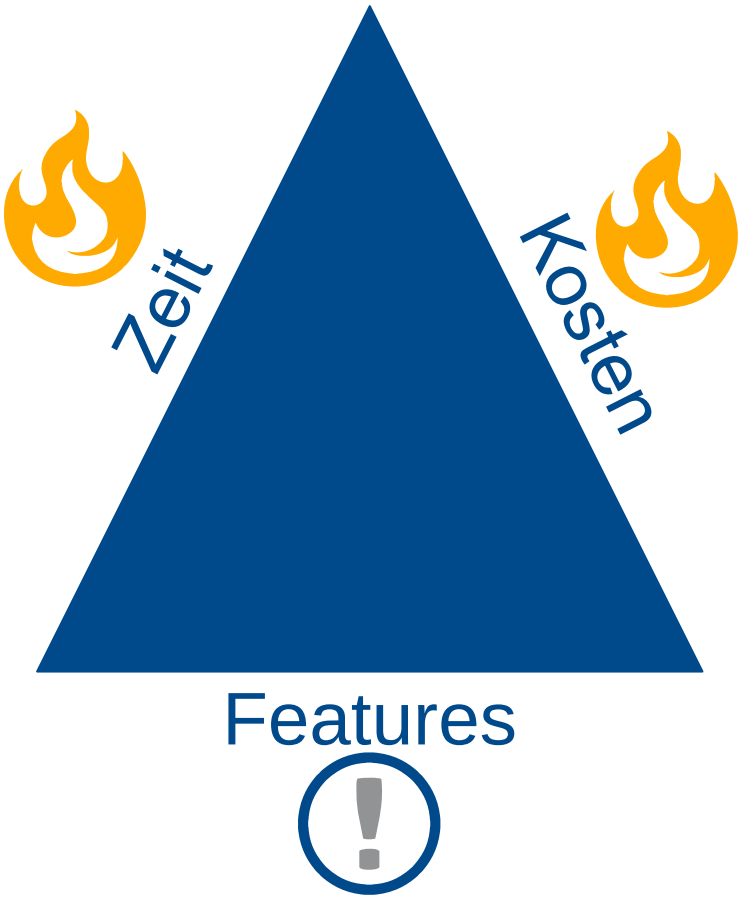
Wir setzen doch „sichere“ Frameworks und Plattformen ein!?

spring

HIBERNATE

und viele andere…

Dokumentation, Tests und Security fallen zuerst weg!

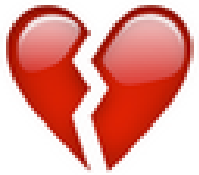


- ~~Dokumentation~~
- ~~Security / Tests~~
- Features!

Entwickler vs. Security!

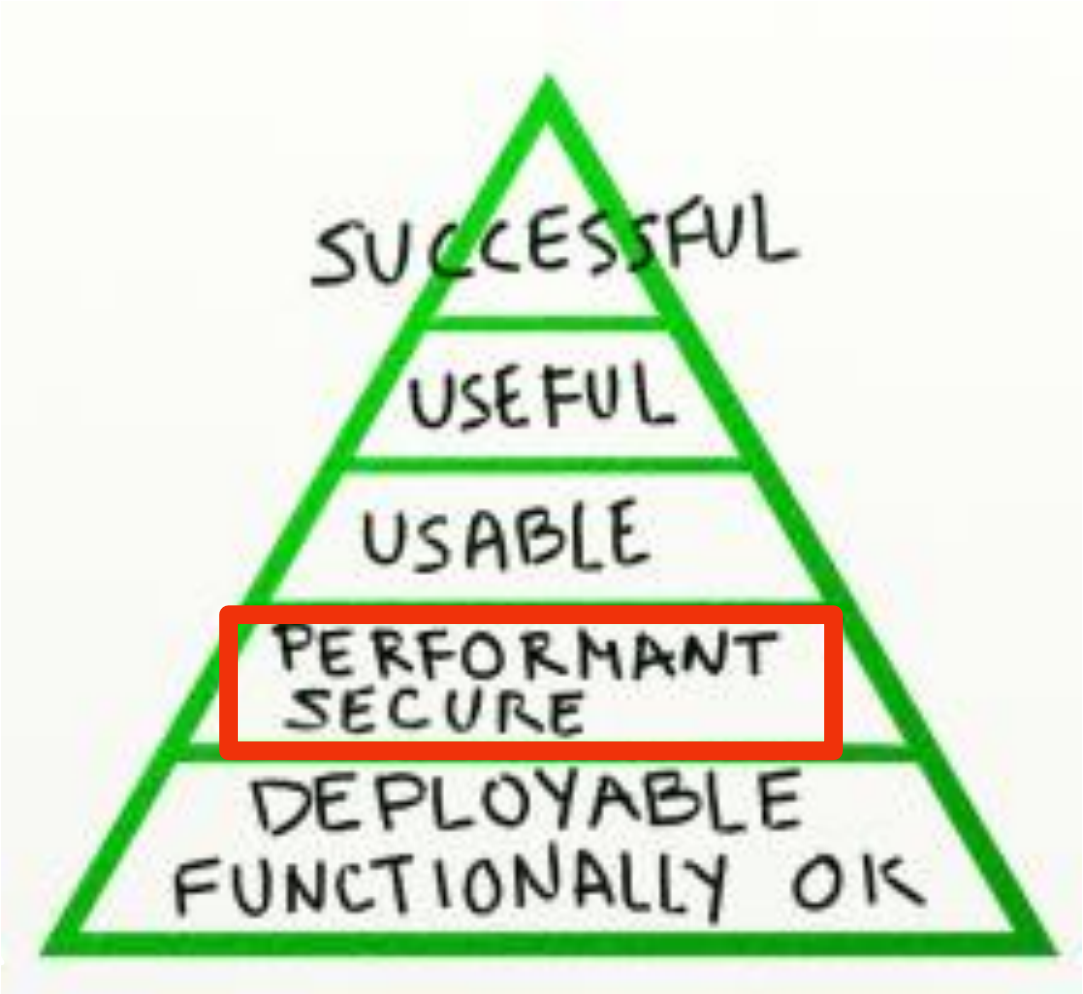







Qualität als Maslow‘sche Pyramide

Maslow‘sche Pyramide



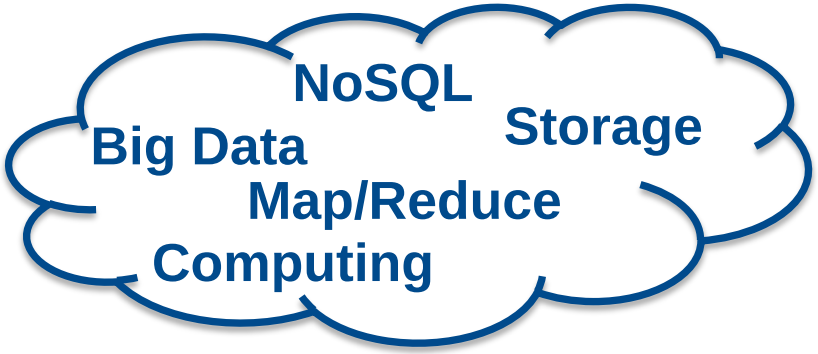
http://gojko.net/2012/05/08/redefining-software-quality

OWASP Top 10 (2013)

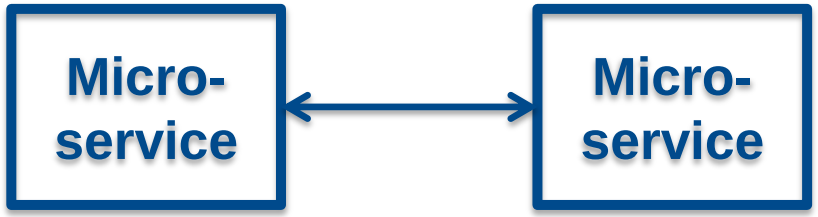
A1 – Injection
A2 – Broken Authentication and Session Management
A3 – Cross-Site Scripting (XSS)
A4 – Insecure Direct Object References
A5 – Security Misconfiguration
A6 – Sensitive Data Exposure
A7 – Missing Function Level Access Control
A8 – Cross-Site Request Forgery (CSRF)
A9 – Using Known Vulnerable Components
A10 – Unvalidated Redirects and Forwards
Merged with 2010-A7 into new 2013-A6

http://owasp.org

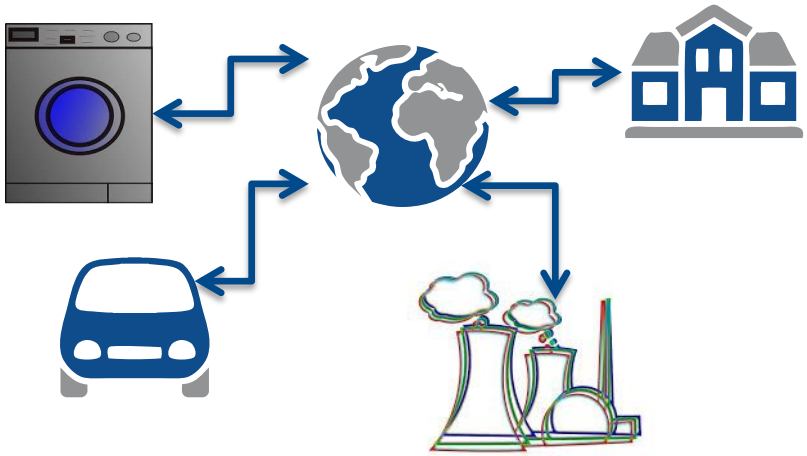
Neue Herausforderungen für Security



Cloud Computing & Big Data

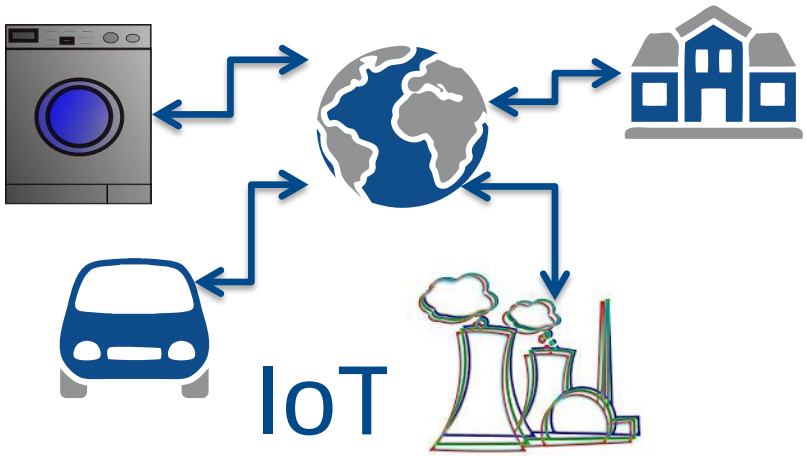


Microservices



Internet of Things (IoT)

Neue Herausforderungen für Security



Updateverteilung

Eigene Backend APIs

3rd Party Backend APIs

Geräte-Interface (Hardware)

Geräte-Interface (Web)

Gerätespeicher

Sensoren

Firmware

Netzwerk-Kommunikation

Cloud

3rd Party Backend APIs

Datenschutz

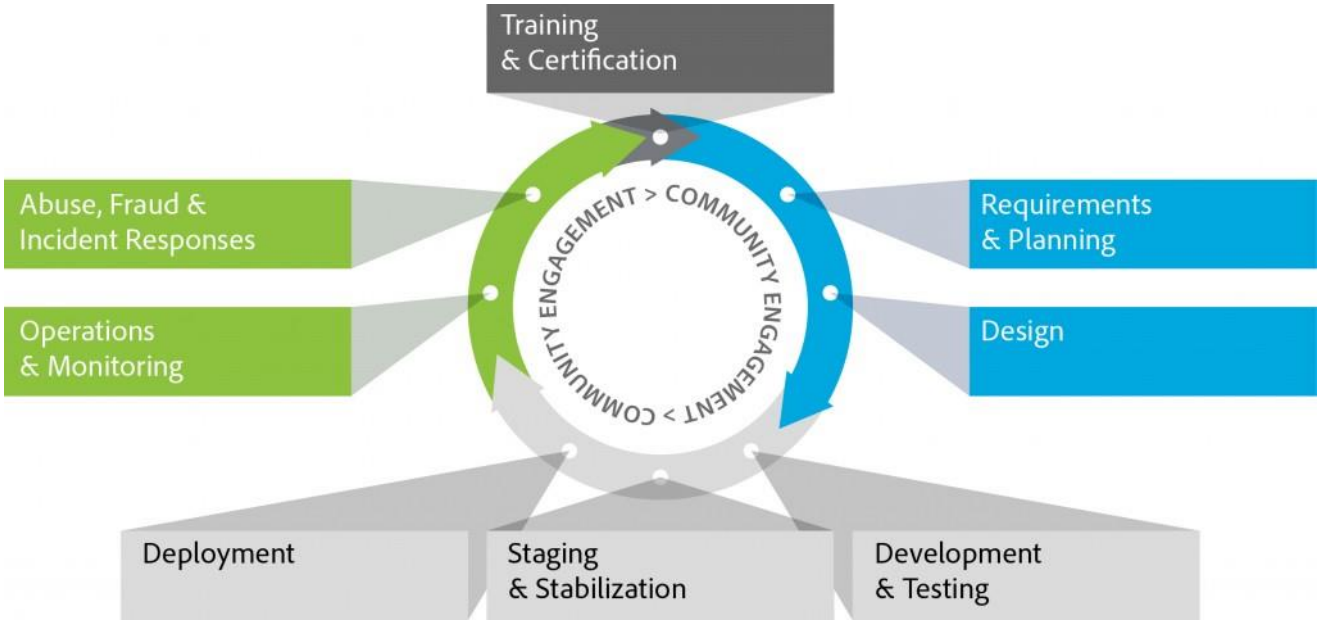
Gibt es nicht schon sichere Entwicklungsprozesse?

Microsoft SDLC

https://www.microsoft.com/en-us/sdl

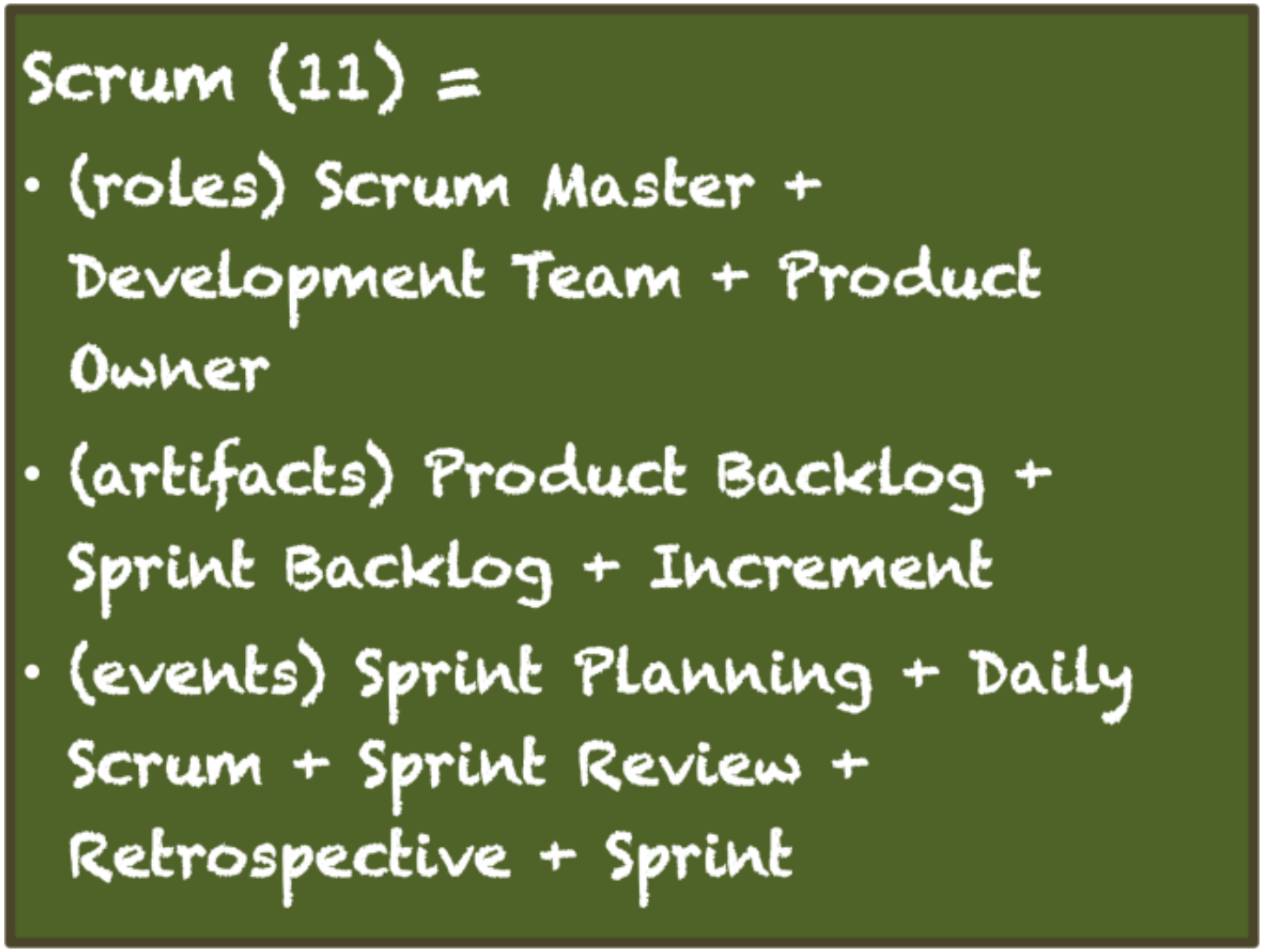
Security @ Adobe

https://www.adobe.com/security/proactive-efforts.html



Next Stop: **Sichere** Agile Entwicklung

Sichere Agile Entwicklung – Scrum Framework Elemente

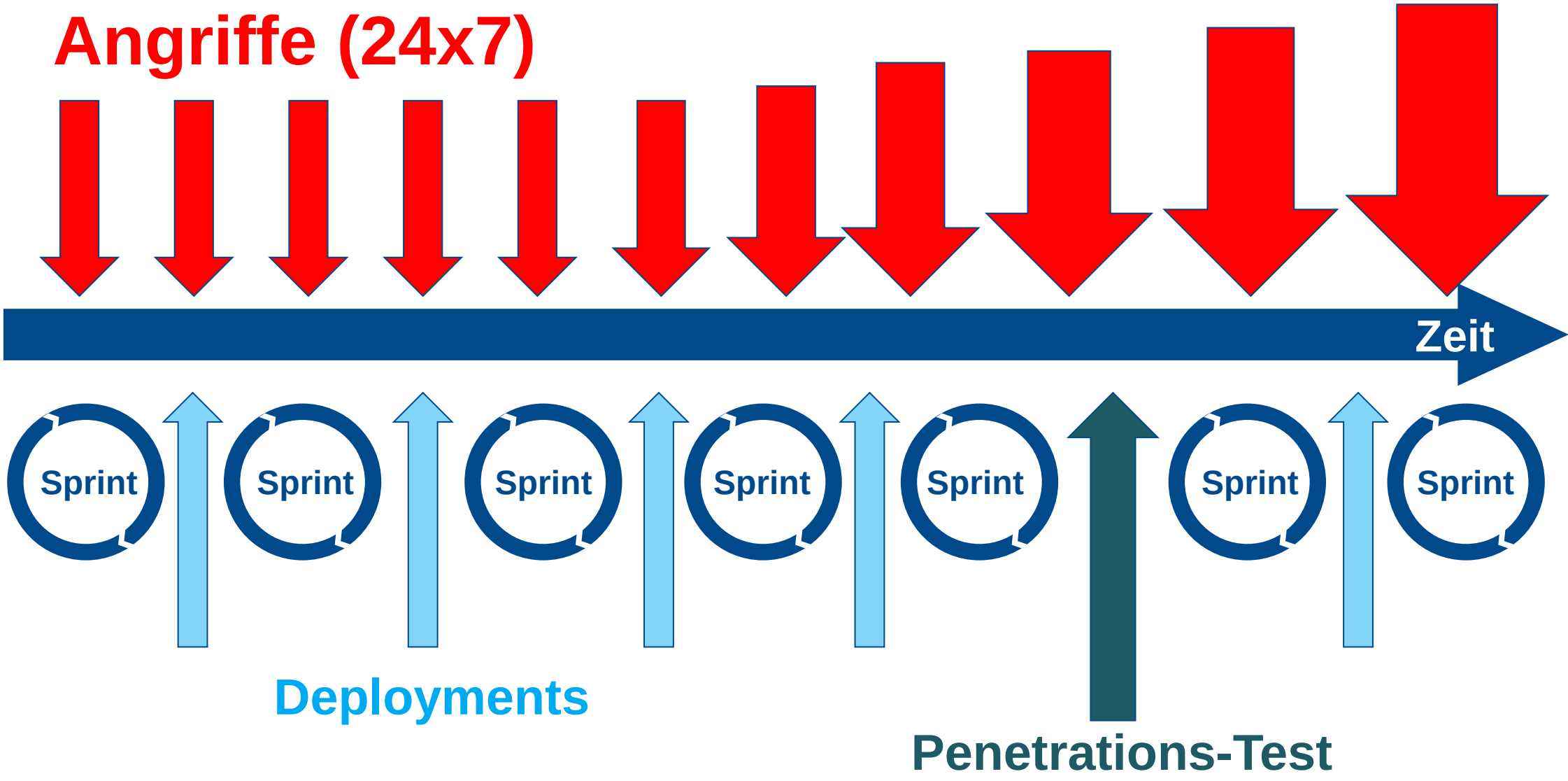


http://guntherverheyen.com/2016/01/29/worrying-interpretations-of-scrum

Ausgangslage: Sicherheit == Agil?

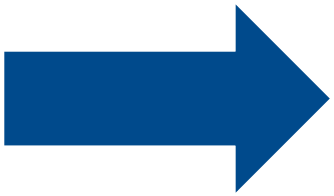
Sprint 1	Sprint 2	Sprint …n	
Story A			
Story B			
	Story C		
	Story D		
	Story E		
		Story F	
		Story G	
		Story H	Go Live
		Security Features	Penetrationstest

Angreifer vs. DevOps



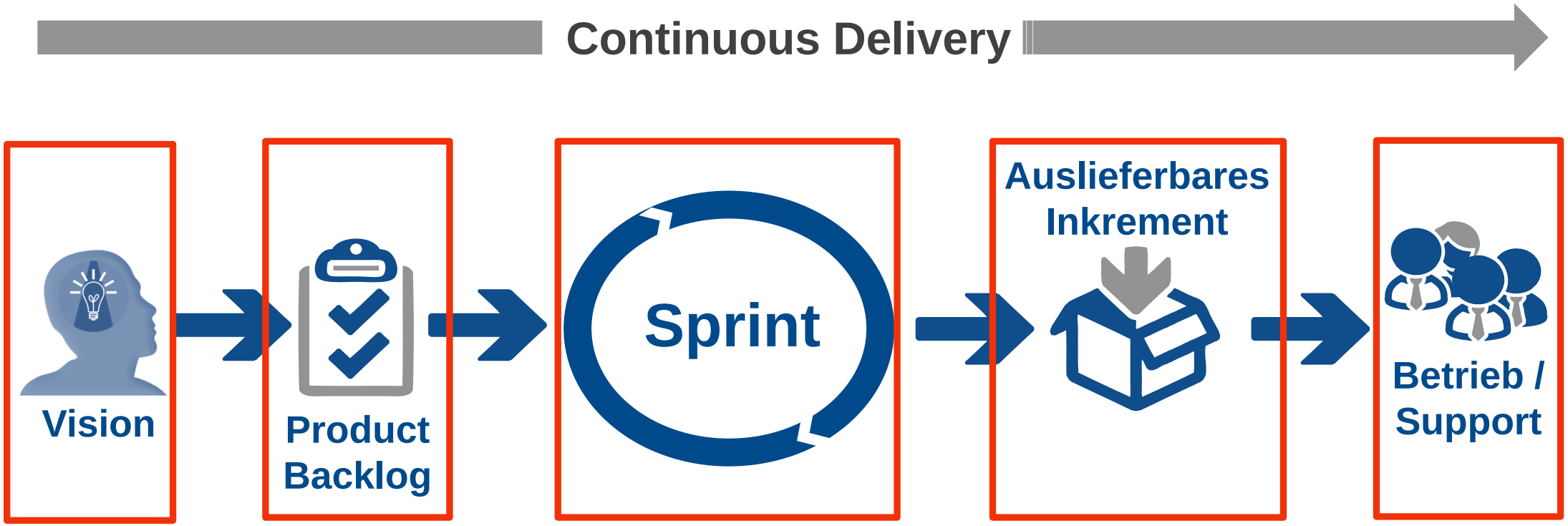
Auslieferbare Inkremente in Scrum

> “Das Entwicklungsteam besteht aus Profis, die am Ende eines jeden Sprints ein fertiges Inkrement übergeben, welches **potentiell auslieferbar** ist.“
>
> http://www.scrumguides.org



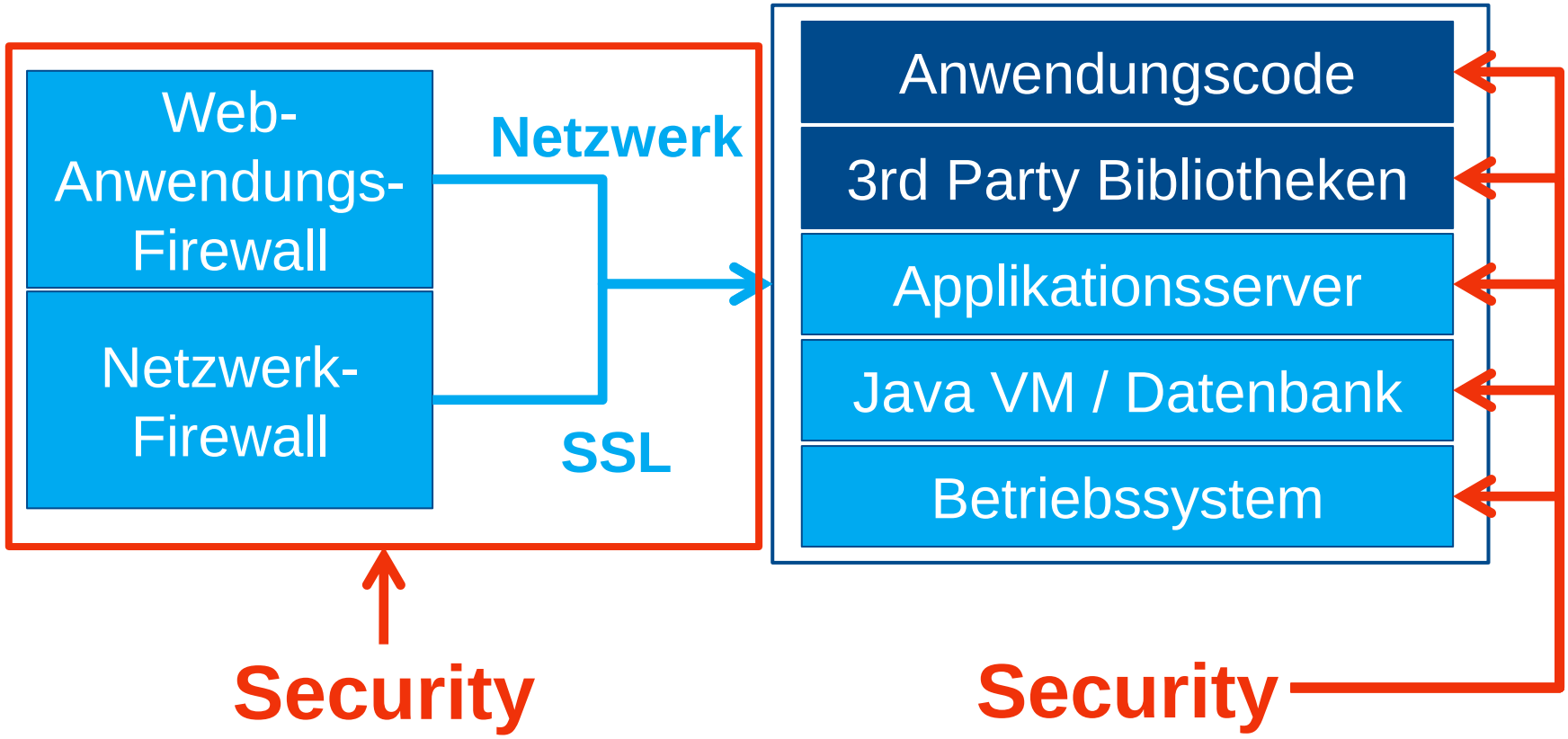
Potentiell **unsicher** ausliefern?

DevOps → SecDevOps

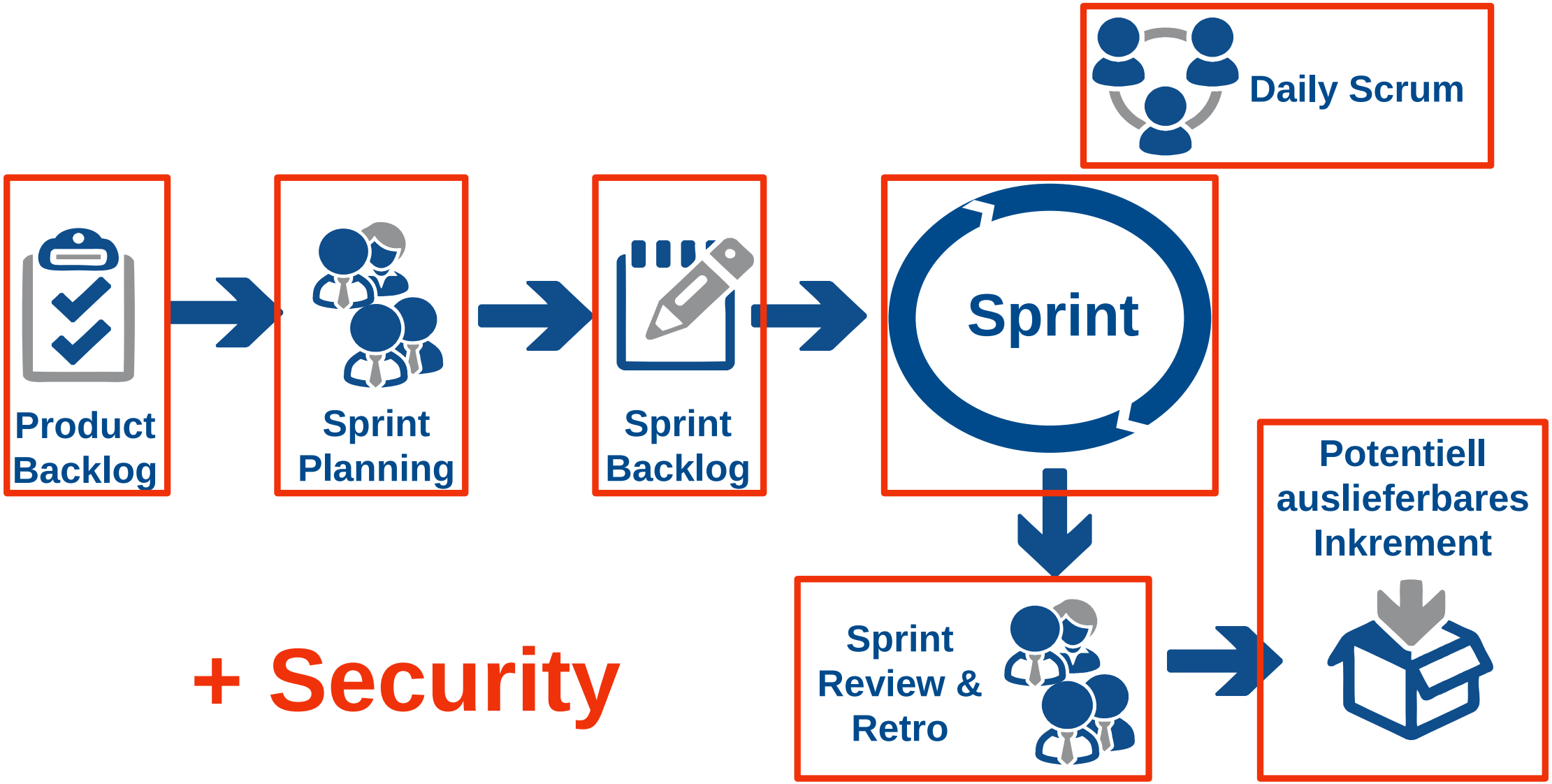


+ Security

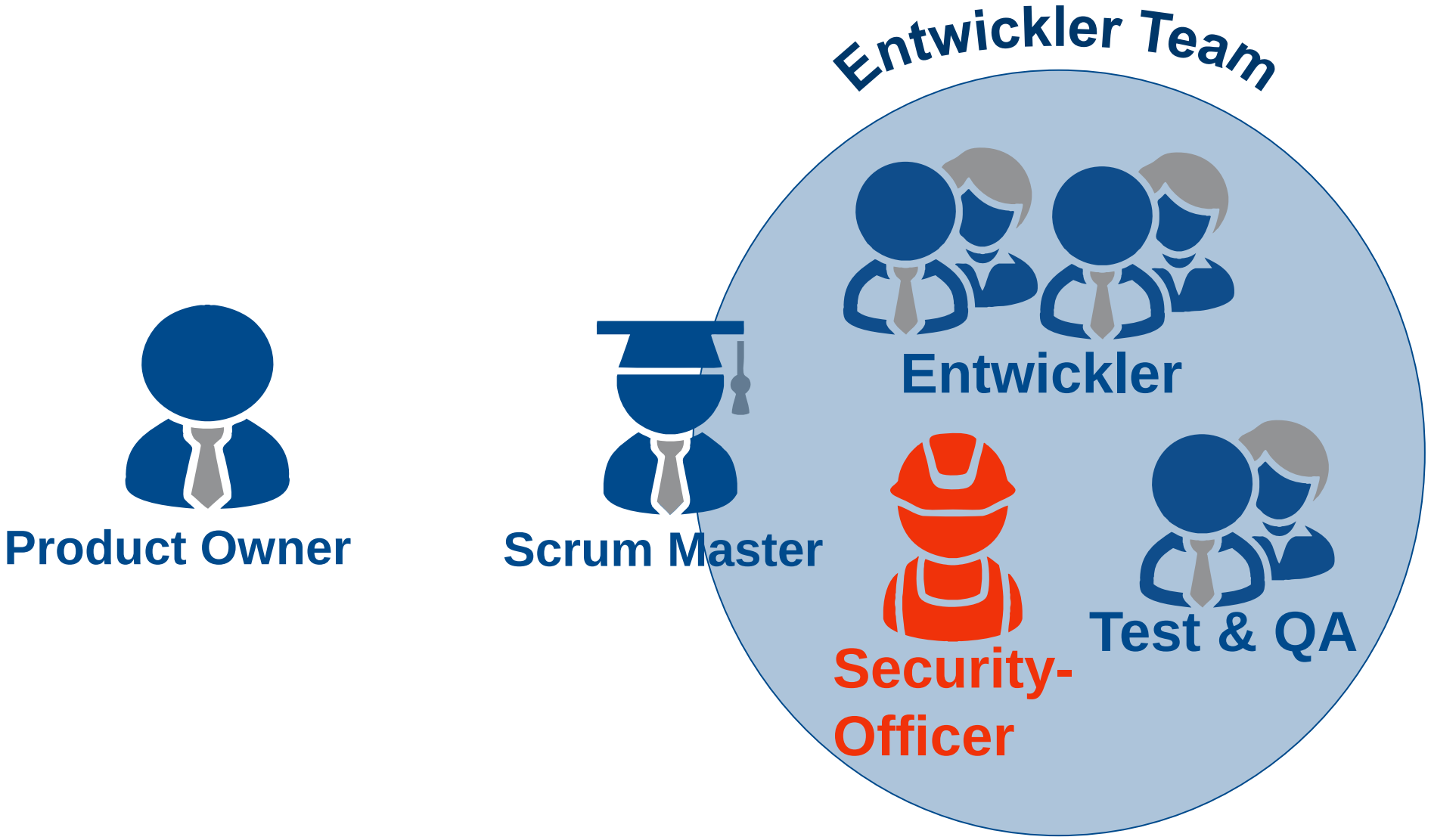
DevOps → **Sec**DevOps



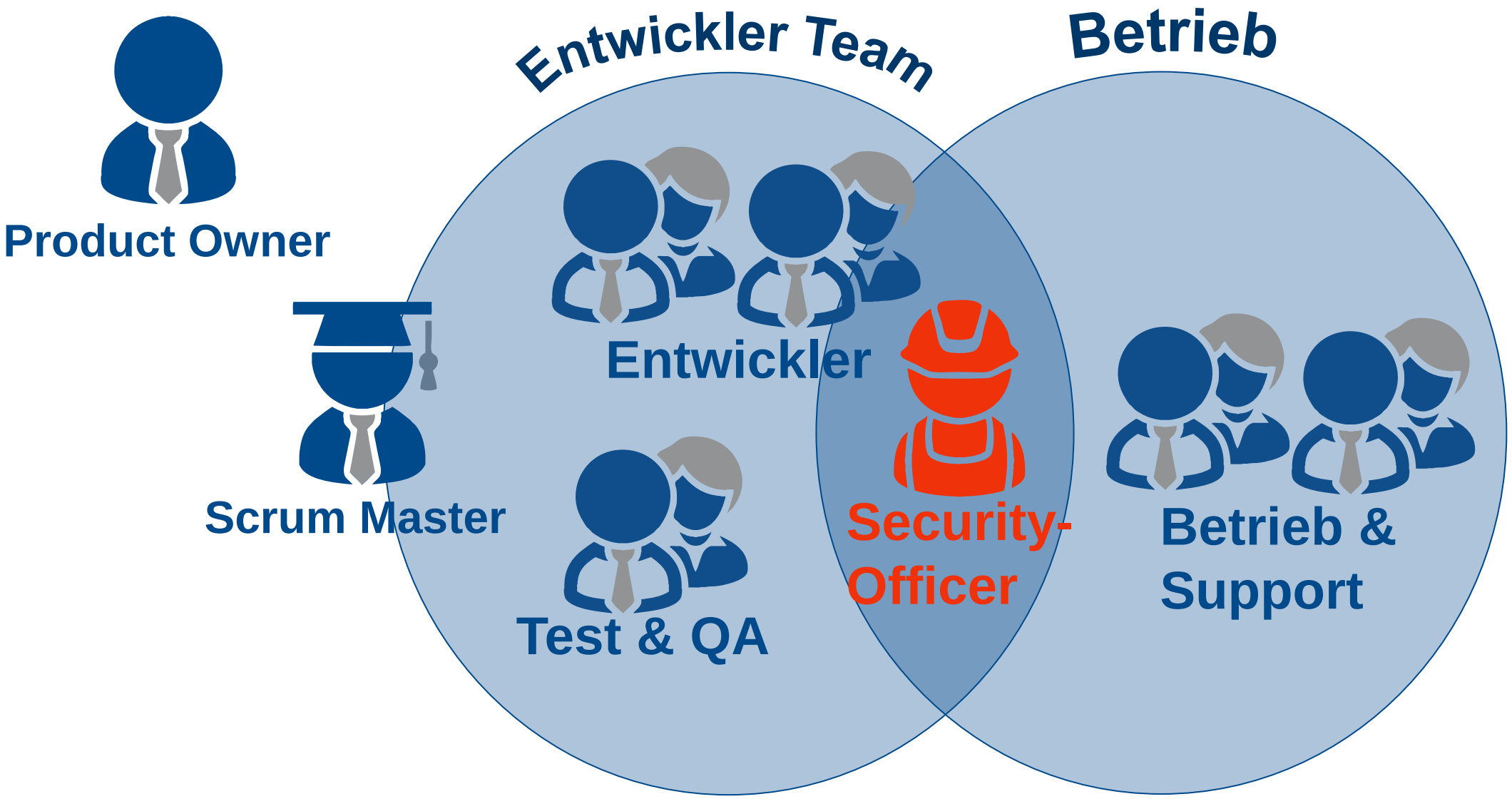
Sichere Agile Entwicklung mit Scrum



Sichere Agile Entwicklung mit Scrum



Secure DevOps = SecDevOps



Rollenspezifische Security Trainings

Security-Officer

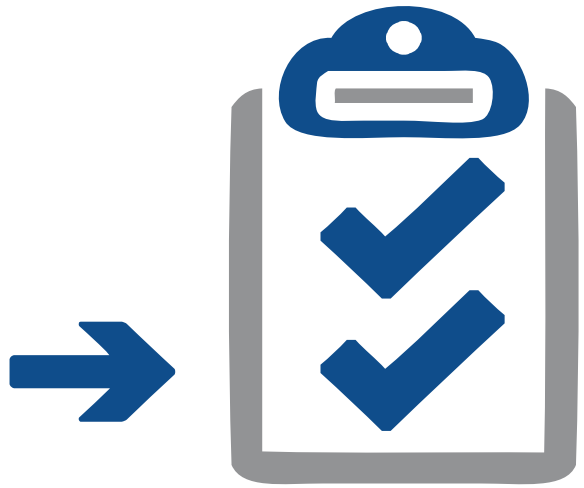
Product Owner

Development Team

Betrieb & Support

Sichere Agile Entwicklung mit Scrum

Threat Modeling



Product Backlog

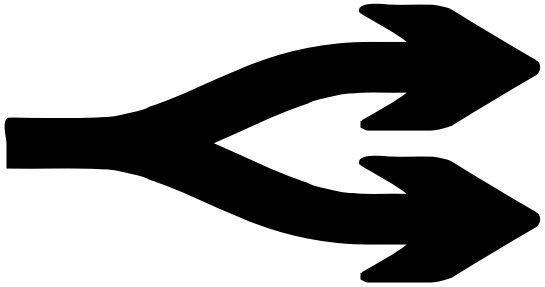
- **Story A**
- **Story B**
- **AbUser Story**
- **Security Features**

Threat Modeling ist auch „Agil“

1. **Festlegung Software-Architektur**
2. **Threat Model**
3. **Identifikation und Vermeidung von Bedrohungen**
4. **Security Testfälle und AbUser Stories**
5. **Test Driven Development (TDD)**
6. **Produktiv Code erstellen**

AbUser Stories

Business User Story



AbUser Story 1

...

AbUser Story N

Als Kunde möchte ich Produkte auswählen und zum Warenkorb hinzufügen um diese zu kaufen.

Als Angreifer möchte ich Anfragen so manipulieren um Preise der Produkte im Warenkorb zu ändern.

Beispiel: **Ab**User und **Security** User Stories

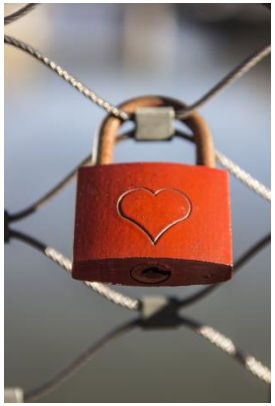
Sichere Agile Entwicklung mit Scrum

To Do	In Progress
TODO-11 TO DO 6 sub-tasks Als Administrator möchte ich Benutzer verwalten um diese für die Anwen	
TODO-10 TO DO 4 sub-tasks Als Script-Kiddie möchte ich mich mit administrativen Rechten anmelden	
TODO-20 ↑ Test auf Session-Fixation (neue JSESSIONID nach Anmeldung)	
TODO-21 ↑ Prüfung, ob Passwort in Klartext ersichtlich ist (UI, Logs, DB, HTTP)	
TODO-22 ↑ Alle Webseiten auf unauthorisierten Zugriff prüfen (Umgehung von Login möglich?)	

Sprint Planning

Sichere Agile Entwicklung mit Scrum

Daily Scrum



Sichere Agile Entwicklung mit Scrum

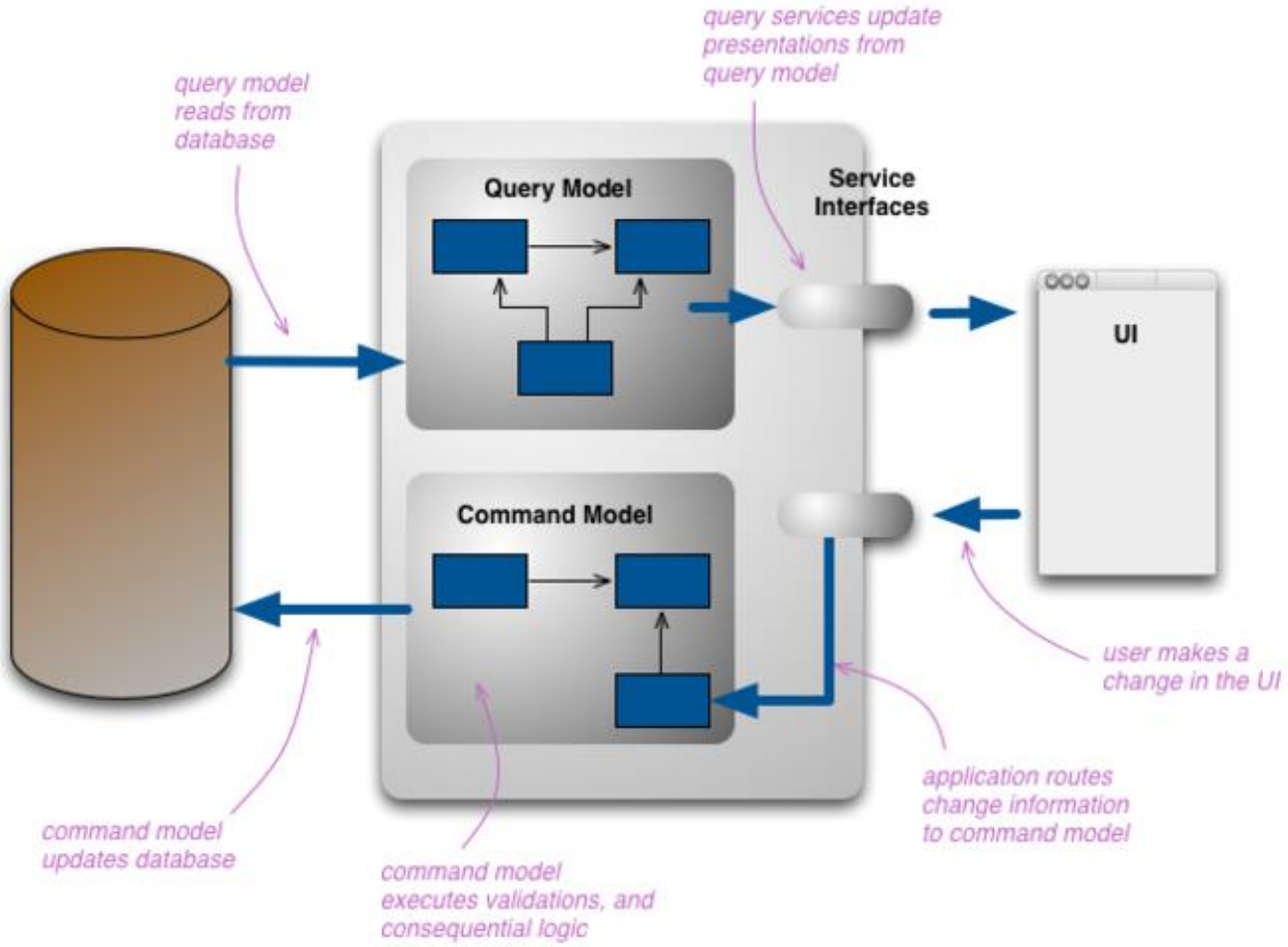
Secure Architektur/Design – Standard Architektur / Patterns

Layering

Microservices

CQRS

MVC



http://martinfowler.com/bliki/CQRS.html

Secure Architektur/Design – Standard Frameworks

Spring Security – **Sichere** “Cloud Native” Anwendungen

Cloud

Single Sign On (SSO)

Spring OAuth2 & Spring SAML

Web

Web Application Security

Spring Security (Core)

„Convention over Configuration“

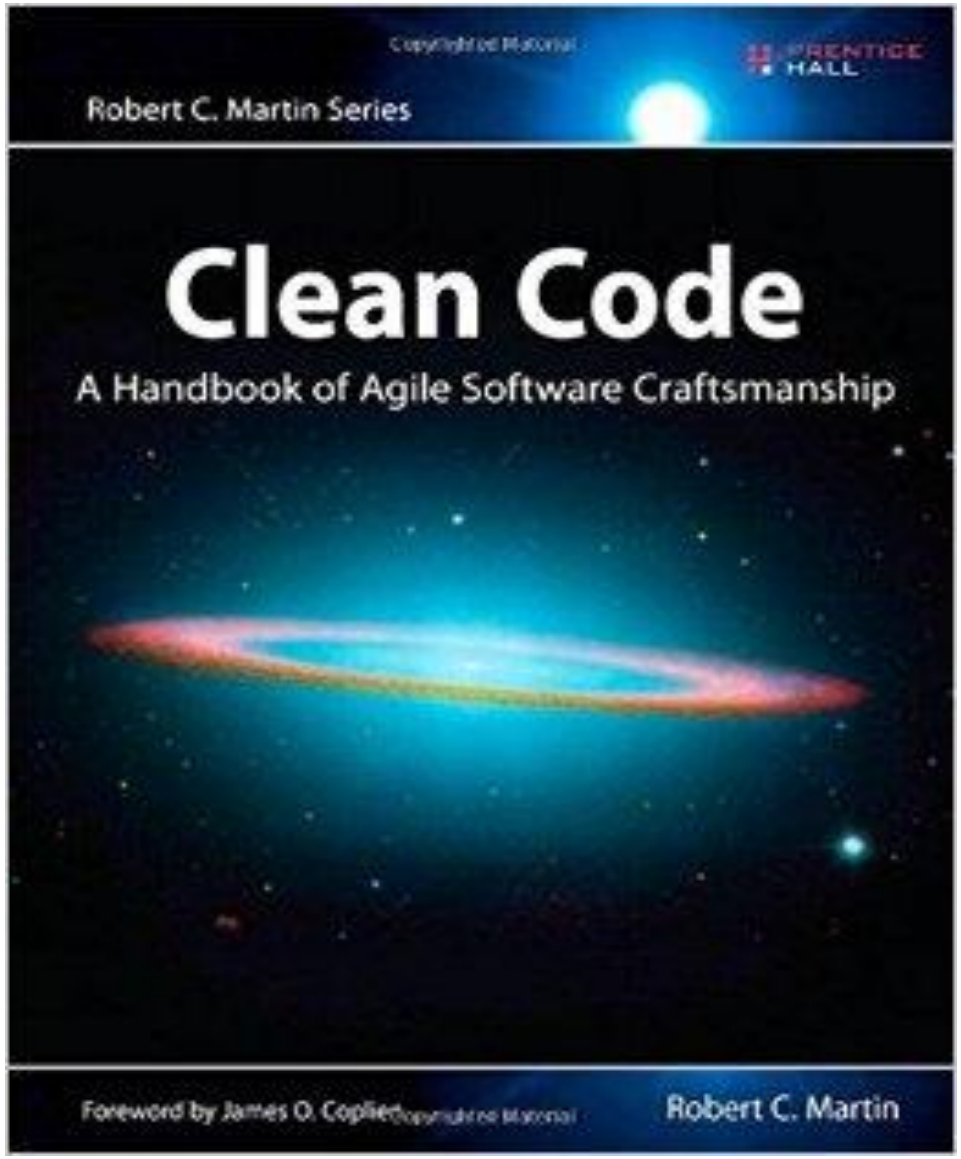
Spring Boot

Enterprise Application Development

Spring IO

Sichere Agile Entwicklung mit Scrum

Secure Design / Coding – Security Patterns

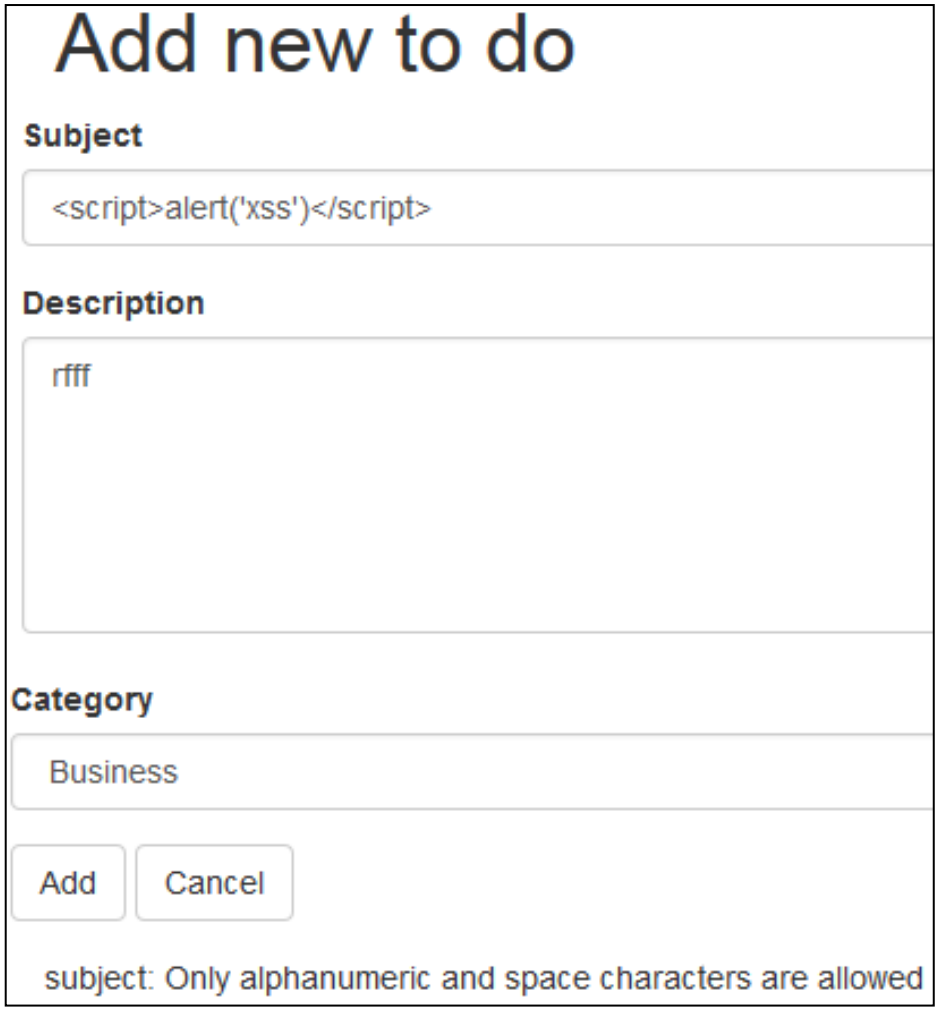


Secure Design / Coding – Cross-Site Scripting

Input Validierung

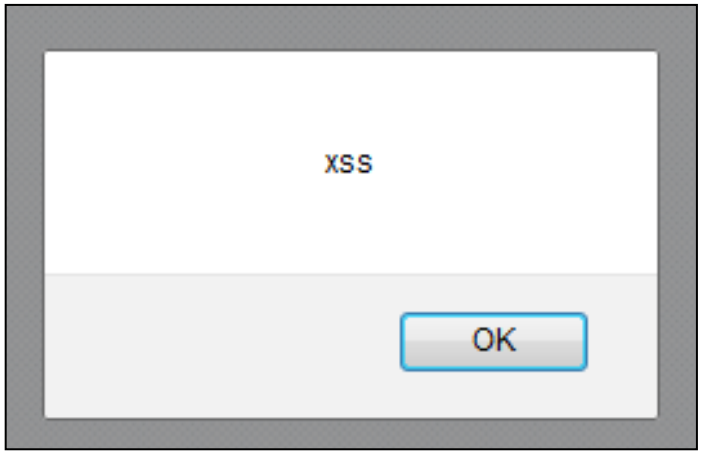
JPA mit „Whitelist“ Bean Validation

```
@NotNull
@Size(min = 1, max = 50)
@Pattern (
    regexp = "^[A-Za-z0-9 ]*$",
    message = "Only alphanumeric and
space characters are allowed" )
private String subject;
```



Secure Design / Coding – Cross-Site Scripting

<script>alert('xss')</script>



Output Escaping

<script>alert('xss')</script>

ToDo's

Index	Subject
1	<script>alert('xss')</script>

Cross-Site Request Forgery (CSRF)

Online Banking Anwendung

GET http://bank.com/transfer.do?acct=BOB&amount=100 HTTP/1.1

Email (HTML-Content)

<img src = "http://bank.com/transfer.do?acct=BOB&amount=100"
width = "0" height = "0" border = "0">

Secure Design / Coding – Security Response Header

▼ **Response Headers** view source

Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Language: de
Content-Length: 394
Content-Security-Policy: default-src 'self'; report-uri /report/
Content-Type: text/html;charset=UTF-8
Date: Tue, 10 May 2016 17:58:35 GMT
Expires: 0
Pragma: no-cache
Public-Key-Pins-Report-Only: max-age=5184000 ; pin-sha256="MIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEAqIlIK+0H
FqGe9xf7HvfpmwisK7gFTwVsrlGU7JGVmPRNdV3RuQl2KFpzwtfmXm0qlwSmsMmkgYqo+IyVYMziQGjZpyHxlg5d96sGTtU4Y+PLSkQJuPQT
wbQ/SpDszgveTVy8WThygMtrN6AcSxYpxk1XBZ6eGsVZD1rgWaIlpfHAPBhH7DALuUDMKf8YYpcYVrViedVMojoAEf1XAE4fyeL3kBBpoKXi
8SPphzfblXW1nDKK7EaDJZxxTqTCA0IK4MCcMXhPaVdEfbpX6fJo8RvXPBxYAORhq92FPS3VUIDDxrQSQdWegADAkHuixJuUikBO92YtXDSu
u0A7vdZ5jMrAr7jbbgyQtZxVcXnEPYTXJOC5fkqfNy3dzqTvfour4sw+cClt/UdgimwOWrKIuEl/SOsaaouQsafPShN4pye2CAe3MqoGQezJ
uH6myDkmMEkJZGXZL9tZX+A4K7vkZYS4aOI3ol1OlmE2fgYh6ANEFFI0ewoW76rvKxI/Ekc2EQW2SeDnXTgOxxG3daJMw+h+tip56dxm9qrw
6/L7D5UodTsNnaqc1ly94ed4qm9MTcErUh8qKzhdSSYOmtemX+iMpPUwFEi9/mEYI2VC+3OmYcVswP42cCAwEAAQ==" ; report-uri="ht
ple.net/pkp-report"
Server: Apache-Coyote/1.1
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
X-XSS-Protection: 1; mode=block

Secure Design / Coding – Sichere Fehlermeldungen

HTTP Status 500 -

type Exception report

message

description The server encountered an internal e

exception

```
org.apache.jasper.JasperException:

11: <title>Urunler</title>
12: </hea
13: <body>
14:     <til     sertDefinition na
15: </body>
16: </htm
```

```
Stacktrace:
        org.apache.jasper.servlet.JspServletWrapper.handleJspException(JspServletWrapper.java:510)
        org.apache.jasper.servlet.JspServletWrapper.service(JspServletWrapper.java:419)
        org.apache.jasper.servlet.JspServlet.serviceJspFile(JspServlet.java:313)
        org.apache.jasper.servlet.JspServlet.service(JspServlet.java:260)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
```

root cause

```
java.lang.NullPointerException
        org.apache.tiles.access.TilesAccess.getContainer(TilesAccess.java:124)
        org.apache.tiles.access.TilesAccess.getContainer(TilesAccess.java:107)
        org.apache.tiles.access.TilesAccess.getCurrentContainer(TilesAccess.java:174)
        org.apache.tiles.template.InsertDefinitionModel.execute(InsertDefinitionModel.java:95)
        org.apache.tiles.jsp.taglib.InsertDefinitionTag.doTag(InsertDefinitionTag.java:254)
```

Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Wed Jan 13 21:48:37 CET 2016
There was an unexpected error (type=Forbidden, status=403).
Access is denied

Secure Design / Coding – Security Patterns

https://www.owasp.org/index.php/OWASP_Proactive_Controls

https://www.owasp.org/index.php/Cheat_Sheets

DEMO

EINE SICHERE WEBANWENDUNG
IN 5 MINUTEN

https://start.spring.io

Sichere Agile Entwicklung mit Scrum

Agile **Security** Testing

PROCESS REVIEWS
& MANUAL INSPECTIONS

CODE REVIEW

SECURITY TESTING

Agile **Security** Testing – Statische Code Analyse

IntelliJIDEA

UNIVERSITY OF MARYLAND 18 56
FindBugs™
because it's easy

+

Find Security Bugs

The FindBugs plugin for security audits of Java web applications.

Agile **Security** Testing – Code Review

Code-Reviews (Github, Gitlab, Gerrit, …)

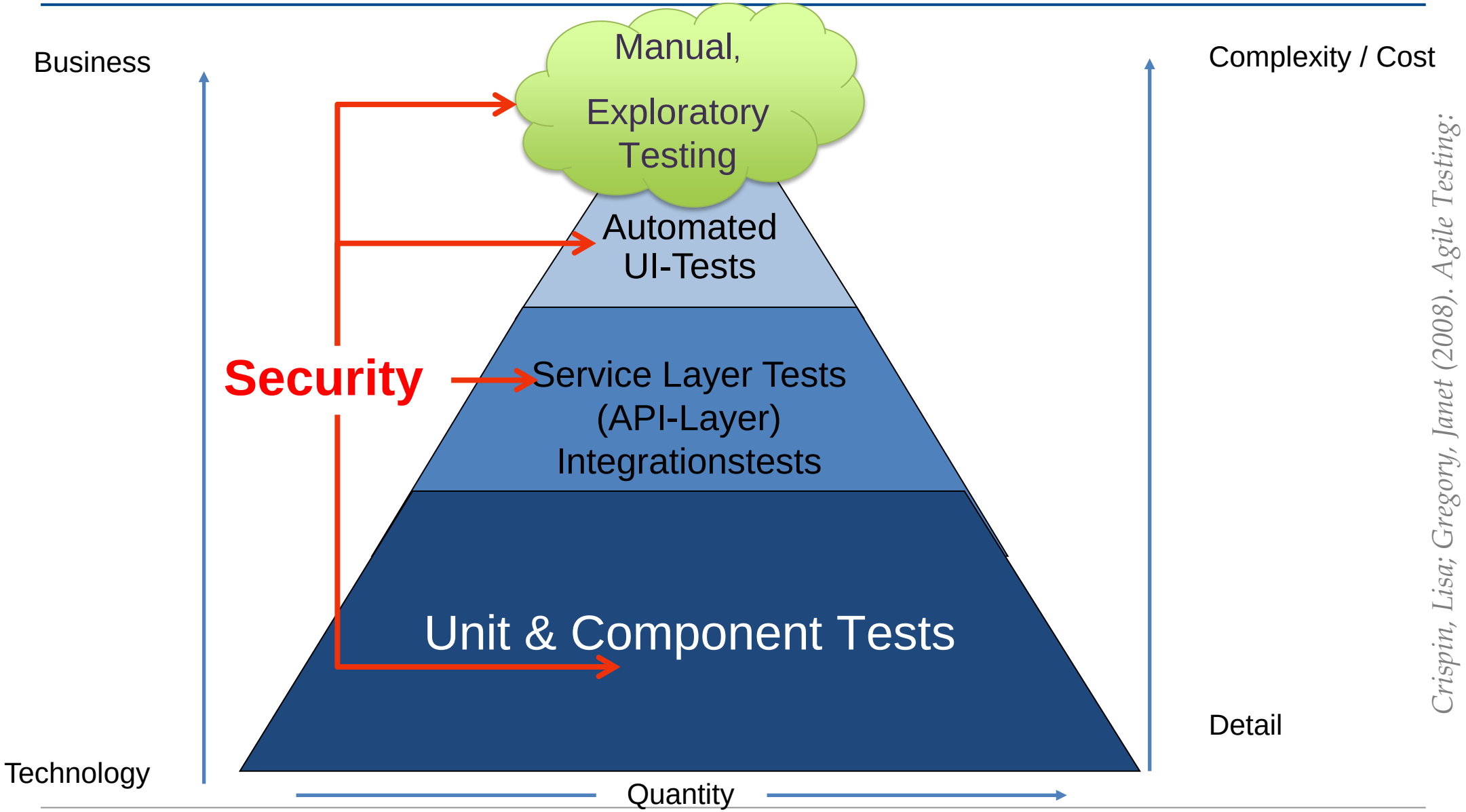
```
89  89
90  90              // Use window.getComputedStyle because jsdom on node.js will break without
91  91              if ( window.getComputedStyle ) {
92      -                   support.pixelPosition = ( window.getComputedStyle( div, null ) || {
93      -                   support.boxSizingReliable = ( window.getComputedStyle( div, null )
    92  +                   divStyle = window.getComputedStyle( div, null ) || { width: "4px"
```

mgol added a note on 15 Apr 2013 Collabo

This adds 5 bytes but prevents `getComputedStyle` to be invoked twice which is nice. **@mikesherov**, what's your opinion?

Add a line note

Agile **Security** Testing



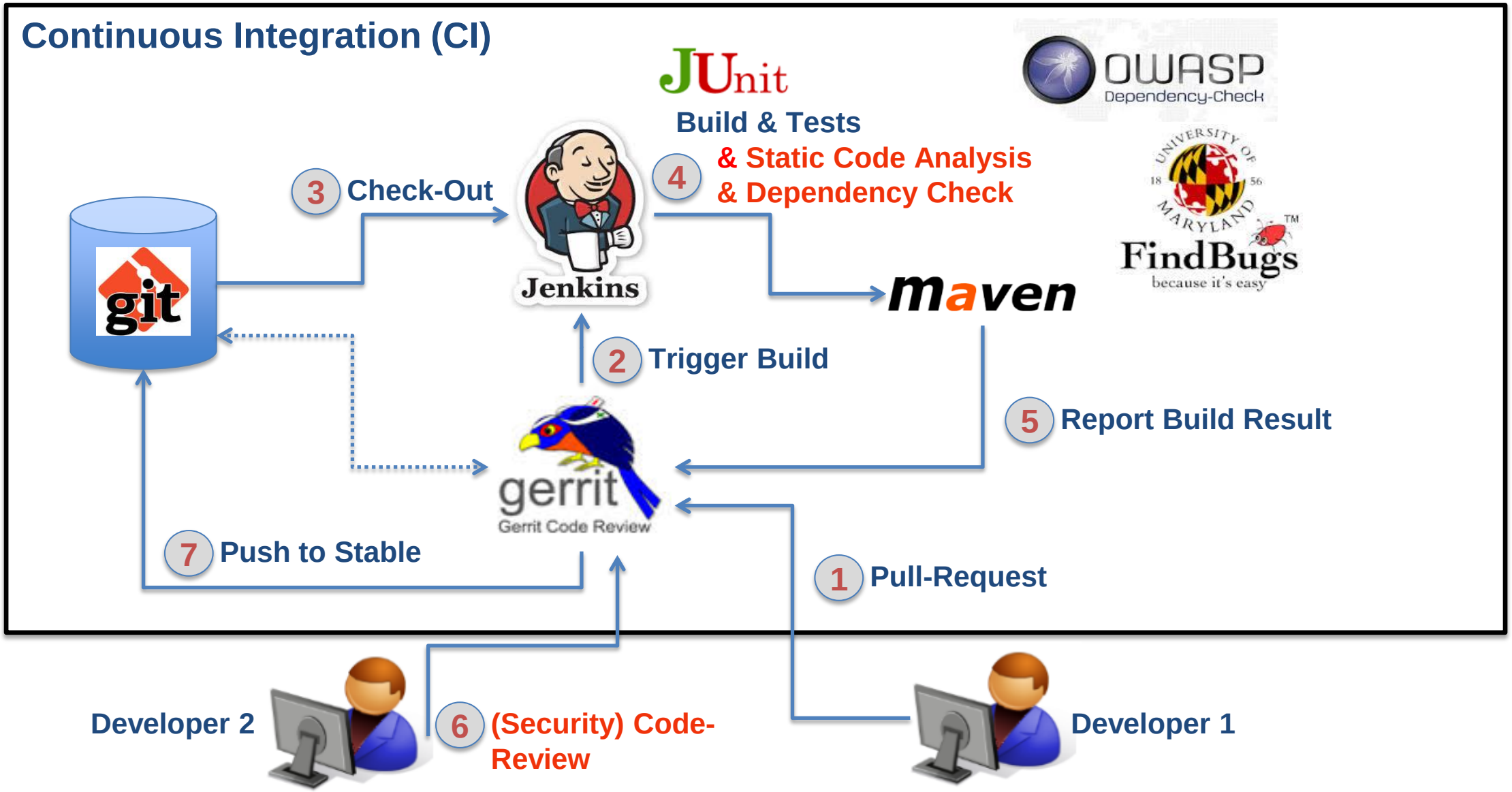
Crispin, Lisa; Gregory, Janet (2008). Agile Testing:

Agile **Security** Testing – Security-Integrationstests

```
@Test
public void verifyAdminPathAuthorizeOK() throws Exception {
    this.mvc.perform( get( "/admin" )
            .with(user("admin").password("admin").roles("ADMIN") ) )
            .andExpect ( status ().isOk () );
}

@Test
public void verifyAdminPathAuthorizeNOK() throws Exception {
    this.mvc.perform ( get( "/admin")
            .with(user("user").password("secure").roles("USER") ) )
            .andExpect ( status ().isForbidden () );
}
```

Stage 1: Statisches Security Testing

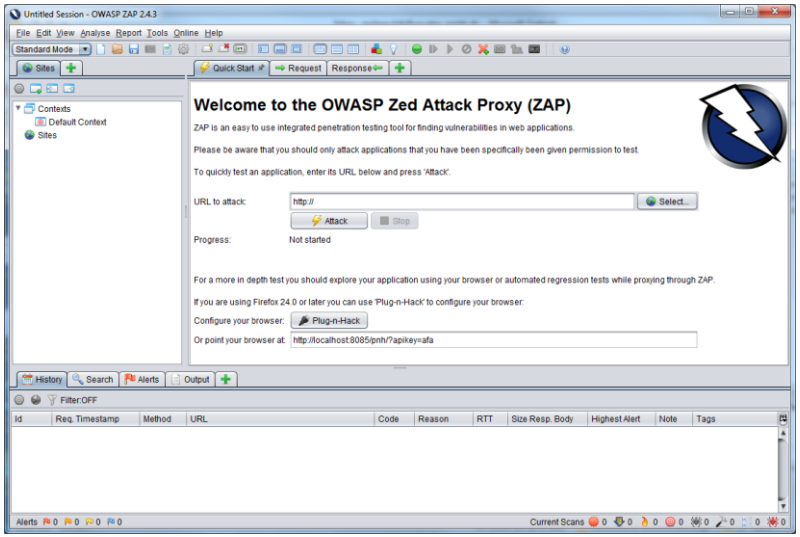


Dynamisches Security-Testing

OWASP Zed Attack Proxy (ZAP)

→ Open Source

https://github.com/zaproxy/zaproxy

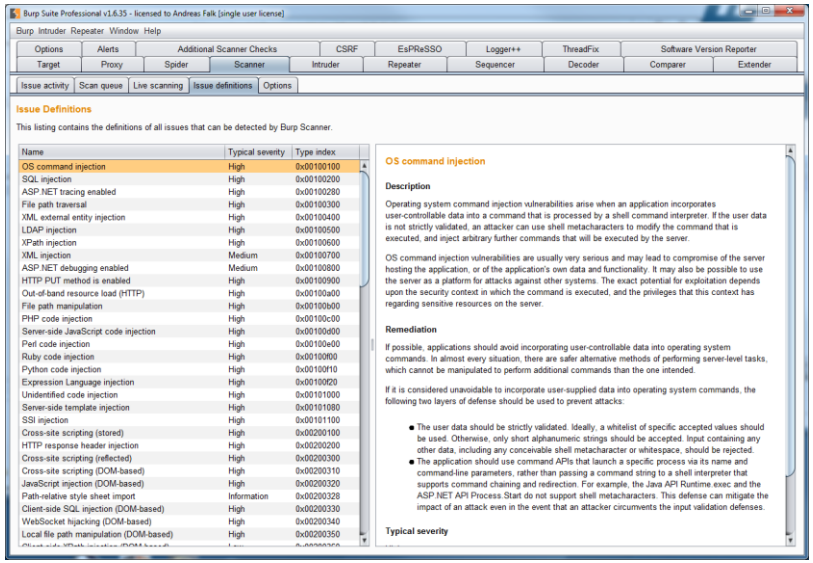


PORTSWIGGER WEB SECURITY

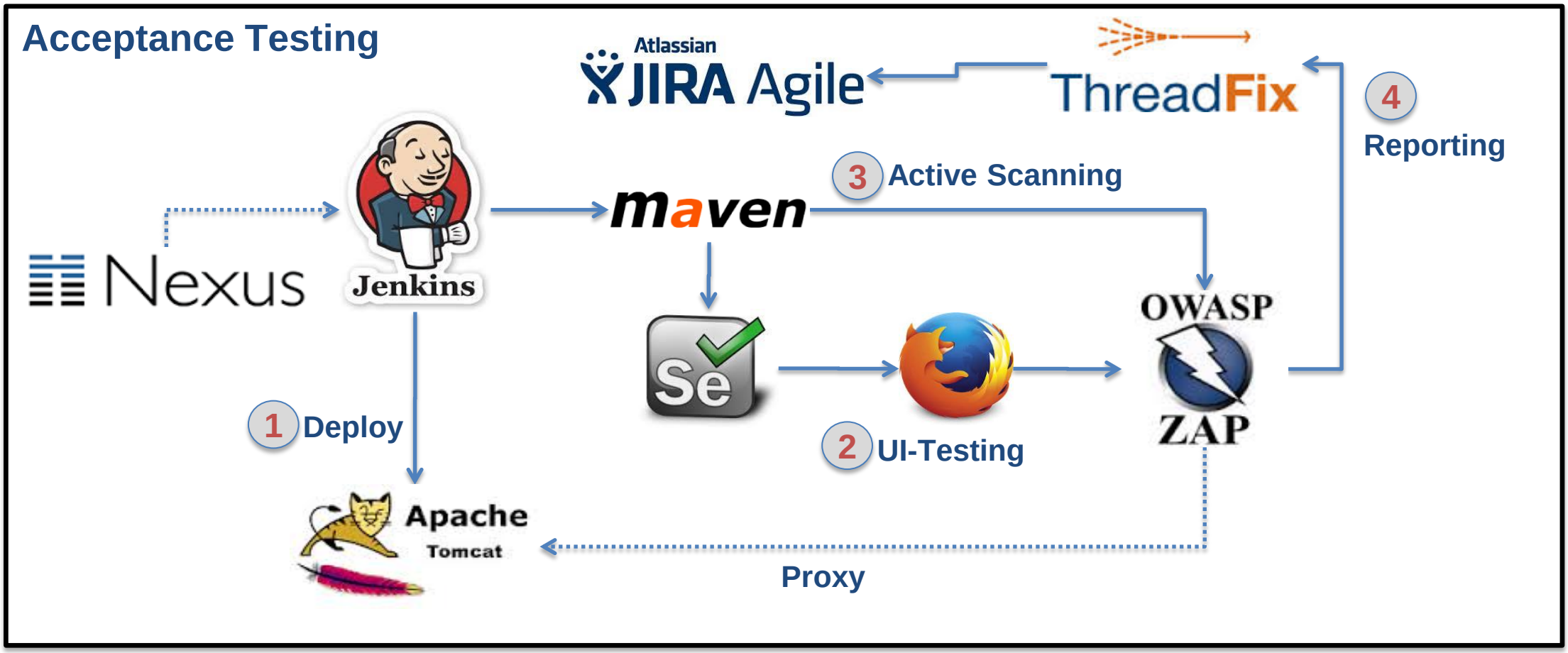
Burp Suite Professional

→ Kommerziell

https://portswigger.net/burp

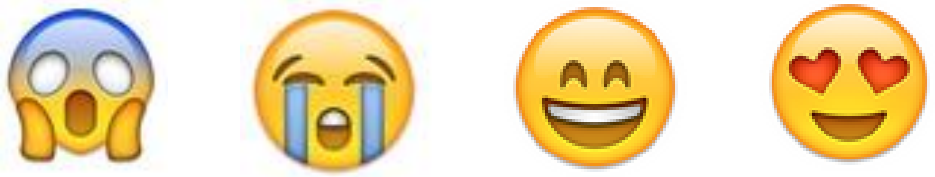


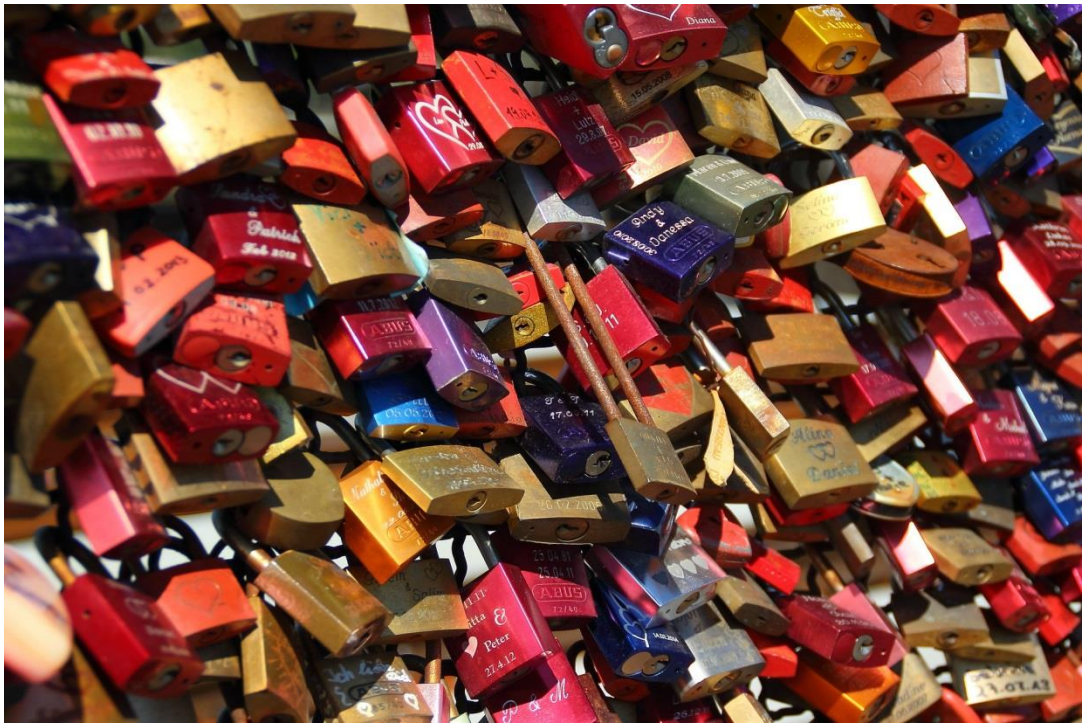
Stage 2: Dynamisches Security-Testing



Sichere Agile Entwicklung mit Scrum

Sprint Review & Retro





Idealzustand: Security == Agile!

Sprint 1	Sprint 2	Sprint …n	
Story A			
Story B			
Abuse Story	Story C		
	Story D		
	Story E		
	Abuse Story	Story F	
Security Features		Story G	
		Story H	Go Live
		Pen-Test	

SecDevOps – Security + DevOps

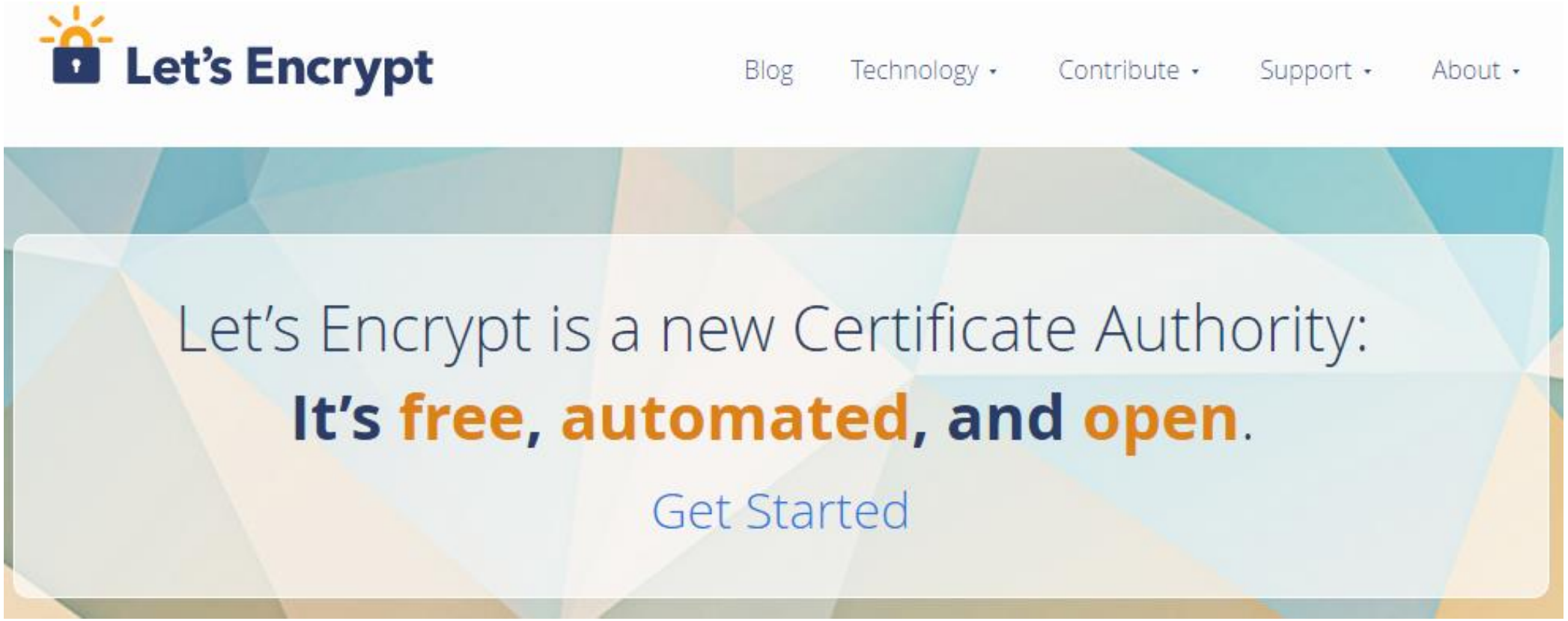
SecDevOps – Testing „Infrastructure As Code“

http://serverspec.org

```
/usr/bin/ruby -S rspec spec/www.example.jp/sample_spec.rb
Package "httpd"  should be installed
Service "httpd"
  should be enabled
  should be running
Port "8443"
  should be listening
Finished in 0.21091 seconds (files took 6.37 seconds to load)
4 examples, 0 failures
```

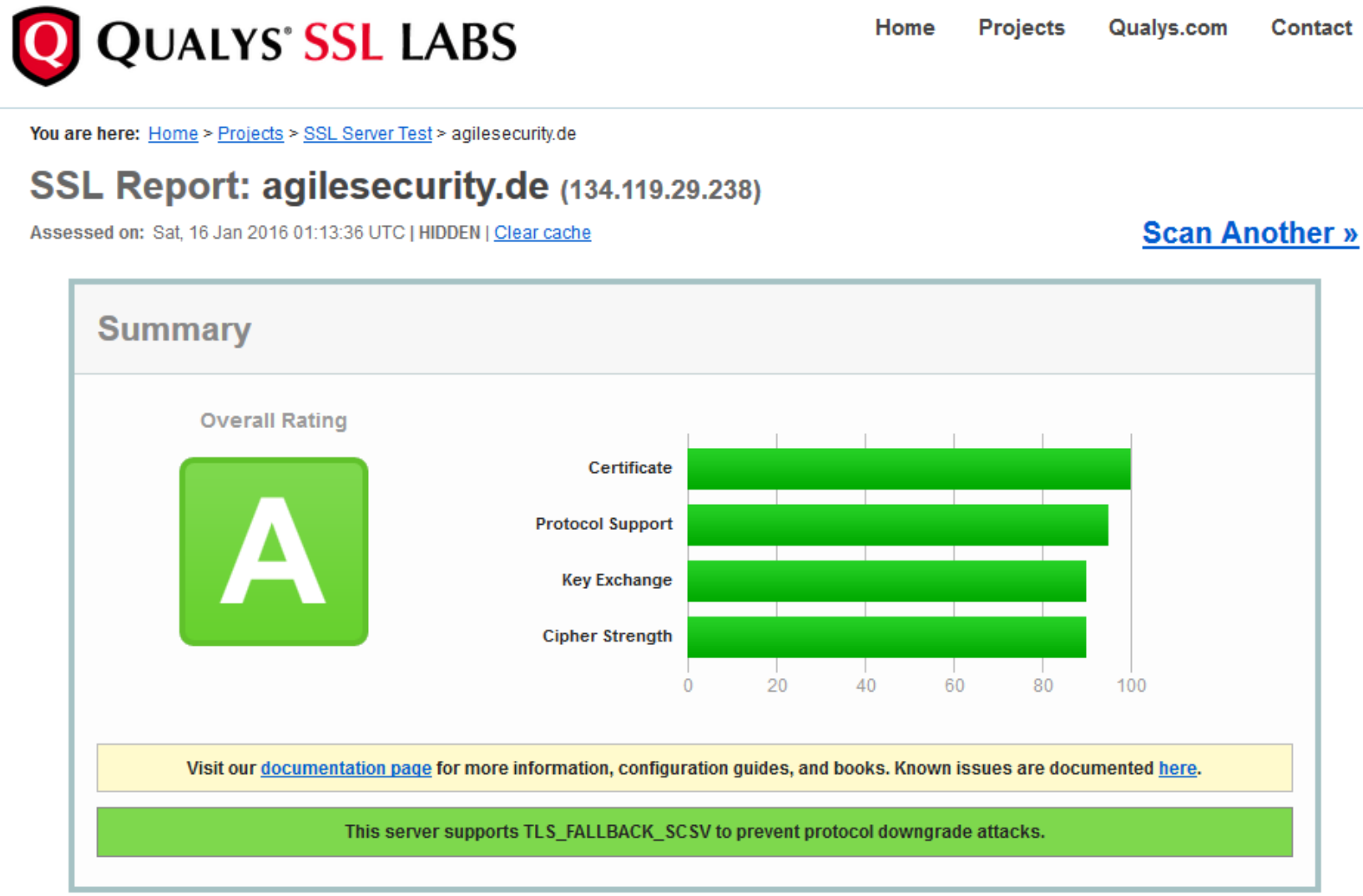
SecDevOps – Kostenlose SSL-Zertifikate

HTTPS für alle Websites !!



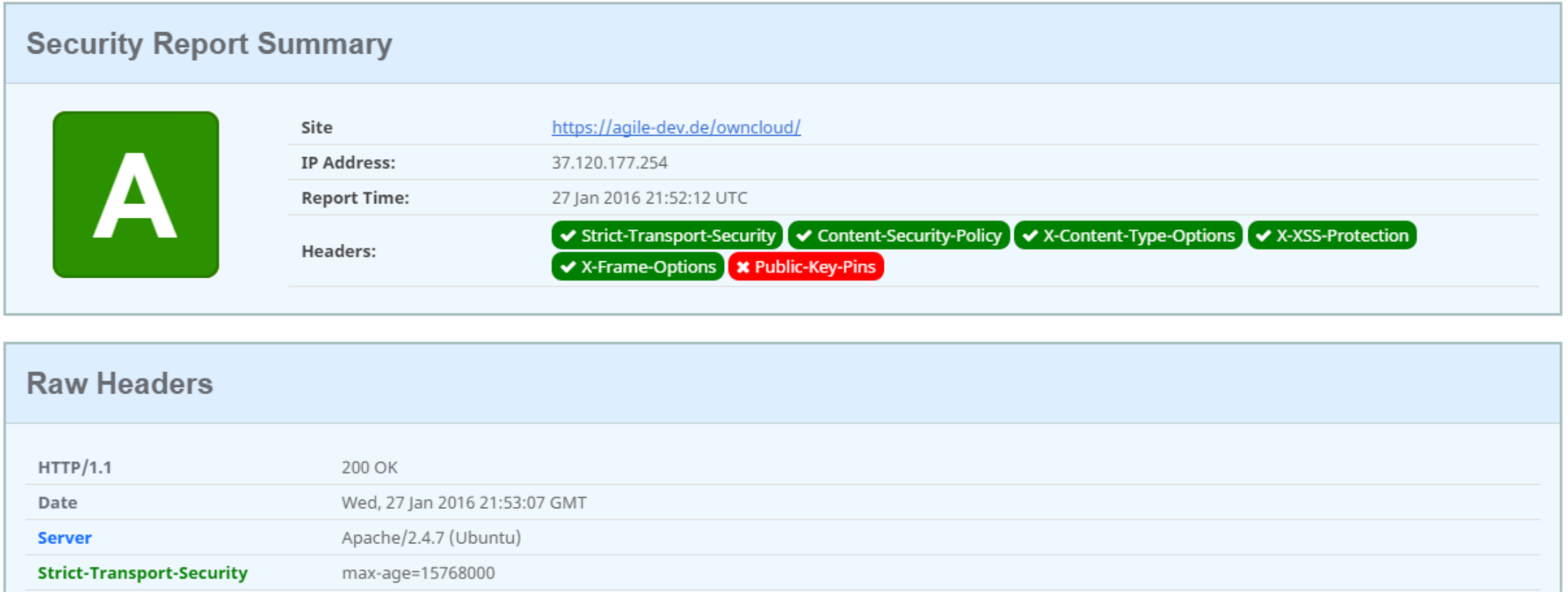
https://letsencrypt.org/

SecDevOps – Sichere TLS (SSL) Konfiguration



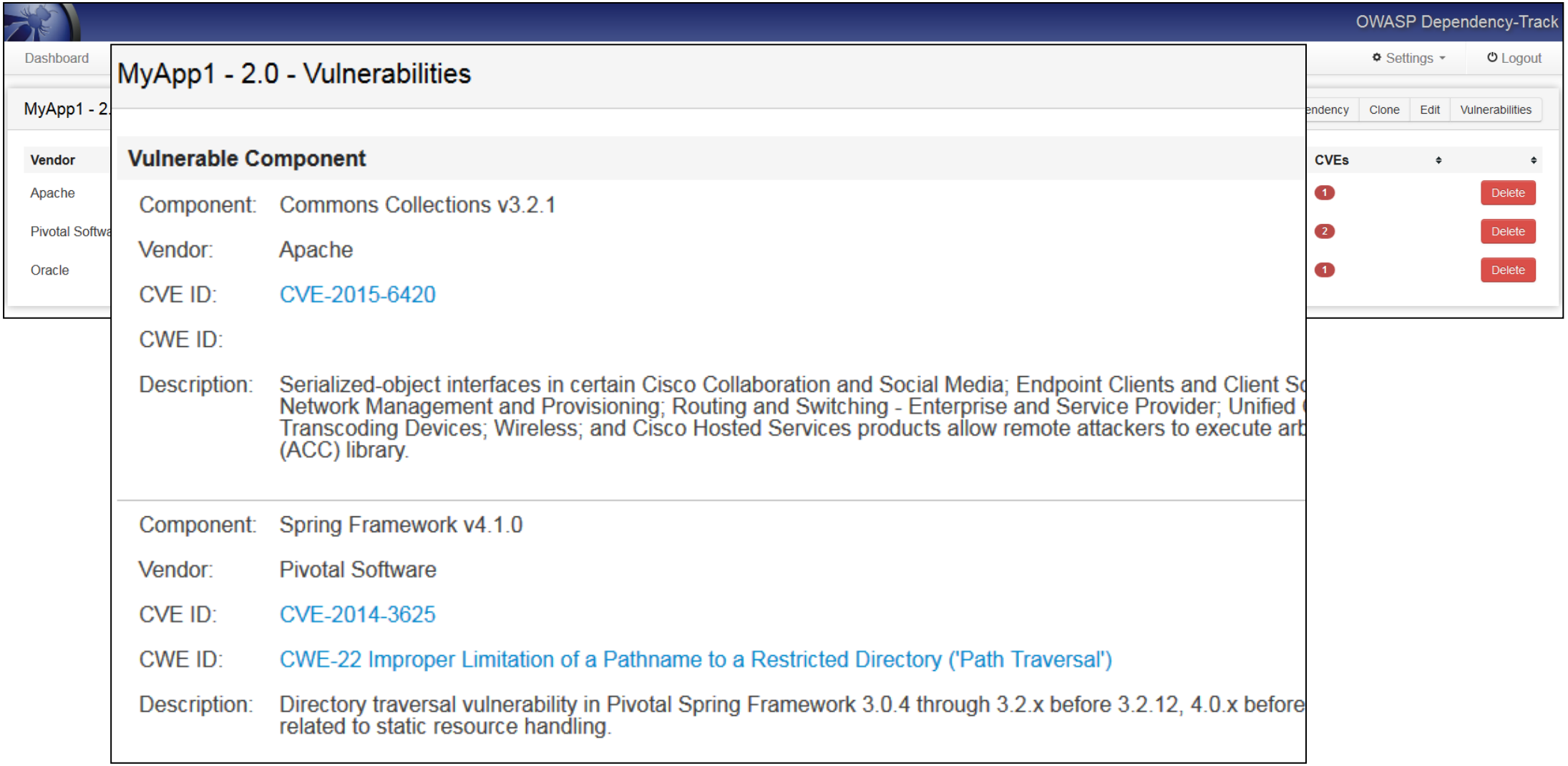
https://www.ssllabs.com/ssltest/index.html

SecDevOps – Sicherheit von HTTP Response-Headern



https://securityheaders.io

SecDevOps – Kenne eingesetzte 3rd Party Bibliotheken



https://www.owasp.org/index.php/OWASP_Dependency_Track_Project

Open **W**eb **A**pplication **S**ecurity **P**roject

Community für sichere Software

Non-Profit!

> 250 lokale „Chapters“

OWASP

Open Web Application Security Project

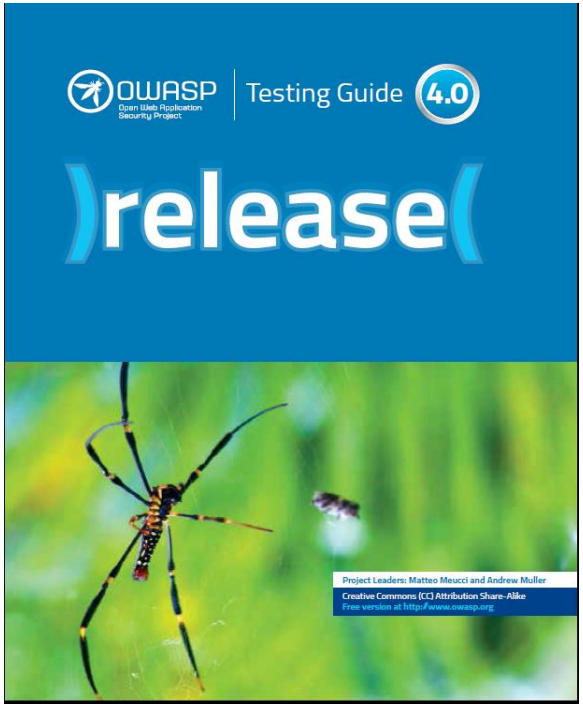
Frei und Offen

> 130 Projekte

> 2500 Mitglieder

https://www.owasp.org

Open **W**eb **A**pplication **S**ecurity **P**roject

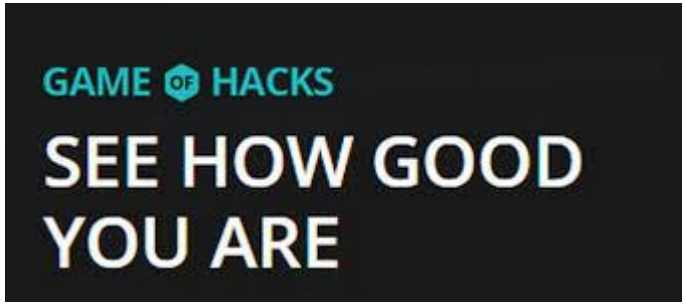


Fazit

Ausbildung für Security

http://www.itsecgames.com

Security transparent machen



http://www.gameofhacks.com

Security-Aktivitäten im gesamten Entwicklungsprozess

Security-Aktivitäten im Projekt

Besuch einer (Security-) Konferenz

https://2016.appsec.eu

Damit das nicht mehr passiert!

FRAGEN?

11.05.2016, ANDREAS FALK

blog.novatec-gmbh.de

@NT_AQE

aqe.novatec-gmbh.de

NOVATEC | EBERHARD KARLS UNIVERSITÄT TÜBINGEN