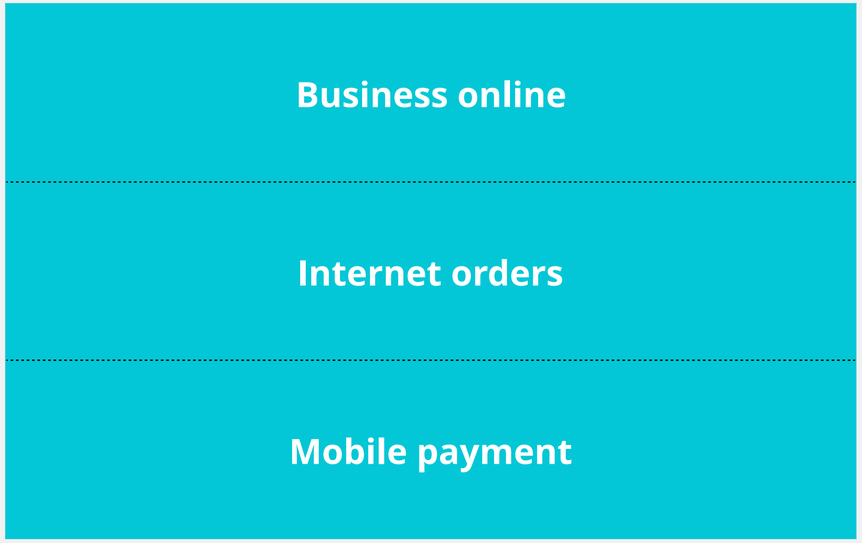
ThoughtWorks®

QA MEET BUILD SECURITY IN DNA

CHANGING

Industry and life are changing

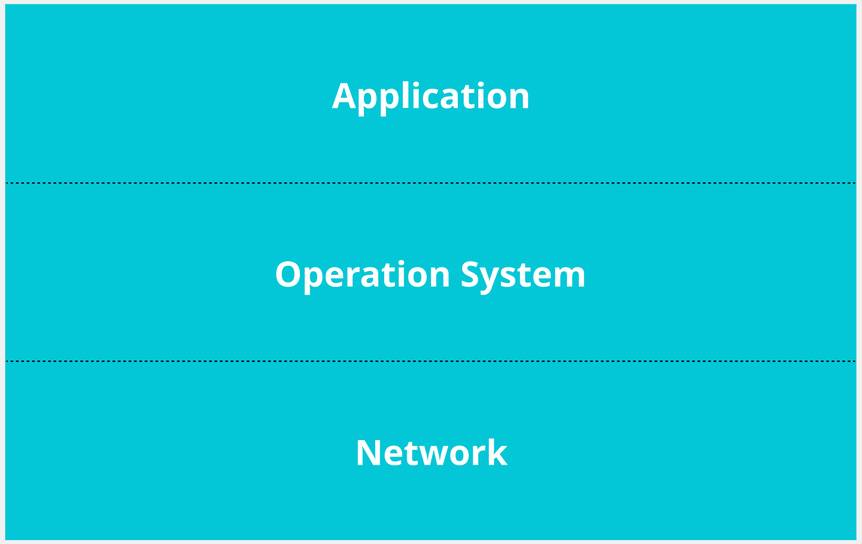


~ 62.2%

~ 216.4%

World Quality Report 2015/16

Threat is changing



~ 80%

World Quality Report 2015/16

BSI OVERVIEW

安全人员培养

注入安全意识和基因，帮助企业培养各个安全角色和安全团队

既有角色

项目 / 技术主管

BA

架构师

程序员

QA

OPS

既有流程

项目规划

需求分析

架构设计

编码实现

测试上线

运维

BSI 流程注入

- 应用安全培训

- 威胁建模

- 软件架构安全
- 拓扑架构安全

- 代码安全审查
- 自动代码安全扫描
- 自动依赖扫描
- 开发工具确认

- 安全扫描
- 安全功能测试
- 自动化安全测试用例

- 定期安全扫描
- 行业安全事件关注

注入安全的行为和基因，帮助建立全生命周期的安全研发流程

BSI PROCESS INJECTION

01. Project Plan

ThoughtWorks® | IT大咖说 知识分享平台

> *This training will raise trai… security awareness and basic security understanding as well as a preparation for the development stage of BSI practice.*

Security Training

Based on OWASP TOP 10, common web security issues, solution and precaution will be introduced with live demos to make every trainee fully understand the security risks.

Tips：The training content covers over 90% of the application security threats.

- *A1 Injection*
- *A2 Broken Authentication and Session Management*
- *A3 Cross-Site Scripting*
- *A4 Broken Access Control*
- *A5 Security Misconfiguration*
- *A6 Sensitive Data Exposure*
- *A7 Insufficient Attack Protection - NEW*
- *A8 Cross-Site Request Forgery*
- *A9 Using Components with Known Vulnerabilities*
- *A10 Underprotected APIs - NEW*

02. Requirement Analysis

ThoughtWorks | IT大咖说 知识分享平台

Threat Modelling

Using attack tree, DREAD and STRIDE model to identify, quantify, and address the security risks associated with an application.

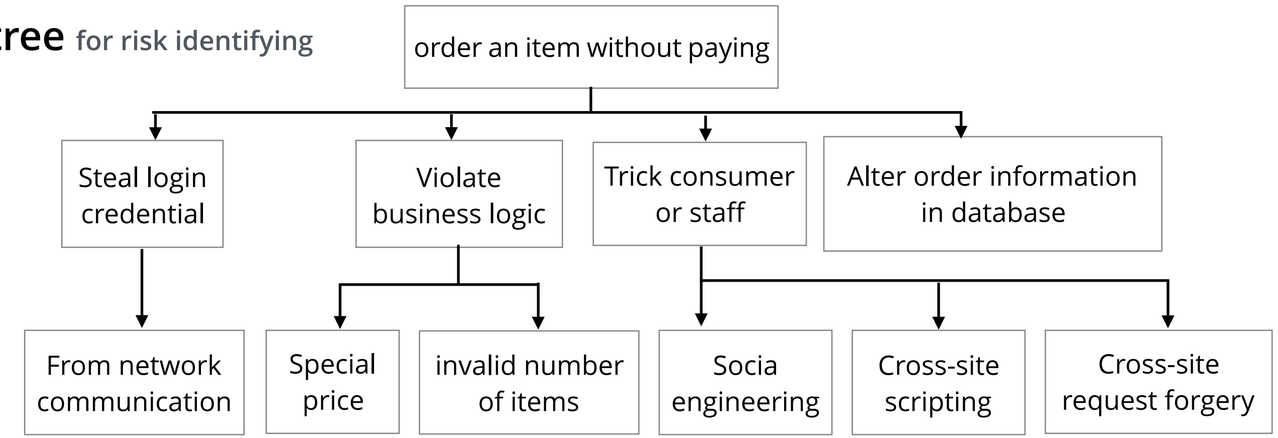
DREAD for risk quantifying

Risk	Questions	Risk Rating
Damage potential	How great is the damage if the vulnerability is exploited?	**High**：12-15 **Medium**：8-11 **Low**：0-7 When a given threat is assessed using DREAD, each category is assigned a value between 1,2 and 3. The sum of all categories for a given exploit can be used to set the risk rating.
Reproducibility	How easy is it to reproduce the attack?	
Exploitability	How easy is it to launch an attack?	
Affected users	How many users are affected?	
Discoverability	How easy is it to find the vulnerability?	

STRIDE for threat identifying

Threat	Description
Spoofing	Forge as another user
Tampering	Malicious modification of data
Repudiation	Denial of the truth of something
Information disclosure	Disclose of information to individuals who aren't supposed to have it
Denial of service	Deny access to valid users
Elevation privilege	Unprivileged user gains privileged access

Attack tree for risk identifying



Example: E-commerce order payment attack tree model

03. Architecture Design

Step 1: Software Architecture Review

- Identify security risks in software architecture

Layered structure or front-end and back-end separating are mostly adopted when software architecture to be designed as scalable and high performance, this step is to check whether all security risks have been fully considered in software architecture design.

Step 2: IT Infrastructure Topology Review

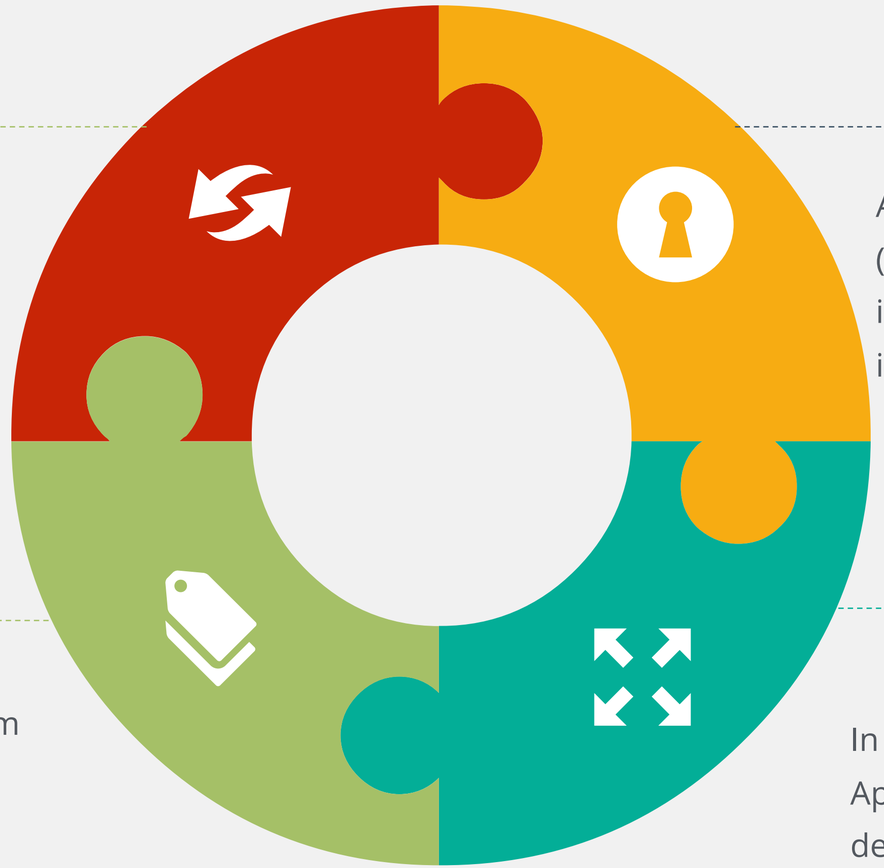
- Identify security risks in IT infrastructure when it carries the software.

Although security is fully considered in software development phase, security risks may be introduced in IT topology design and implementation.

04. Development

Manual security review for source code

Based on secure programming principles, to perform manual security review and improve secure programming awareness.



Automatic source code scan

Apply automatic source code static scan via tools (Fortify, Findbugs, Clockwork), some security issues introduced by inappropriate coding can be quickly identified then fixed.

Automatic dependencies scan

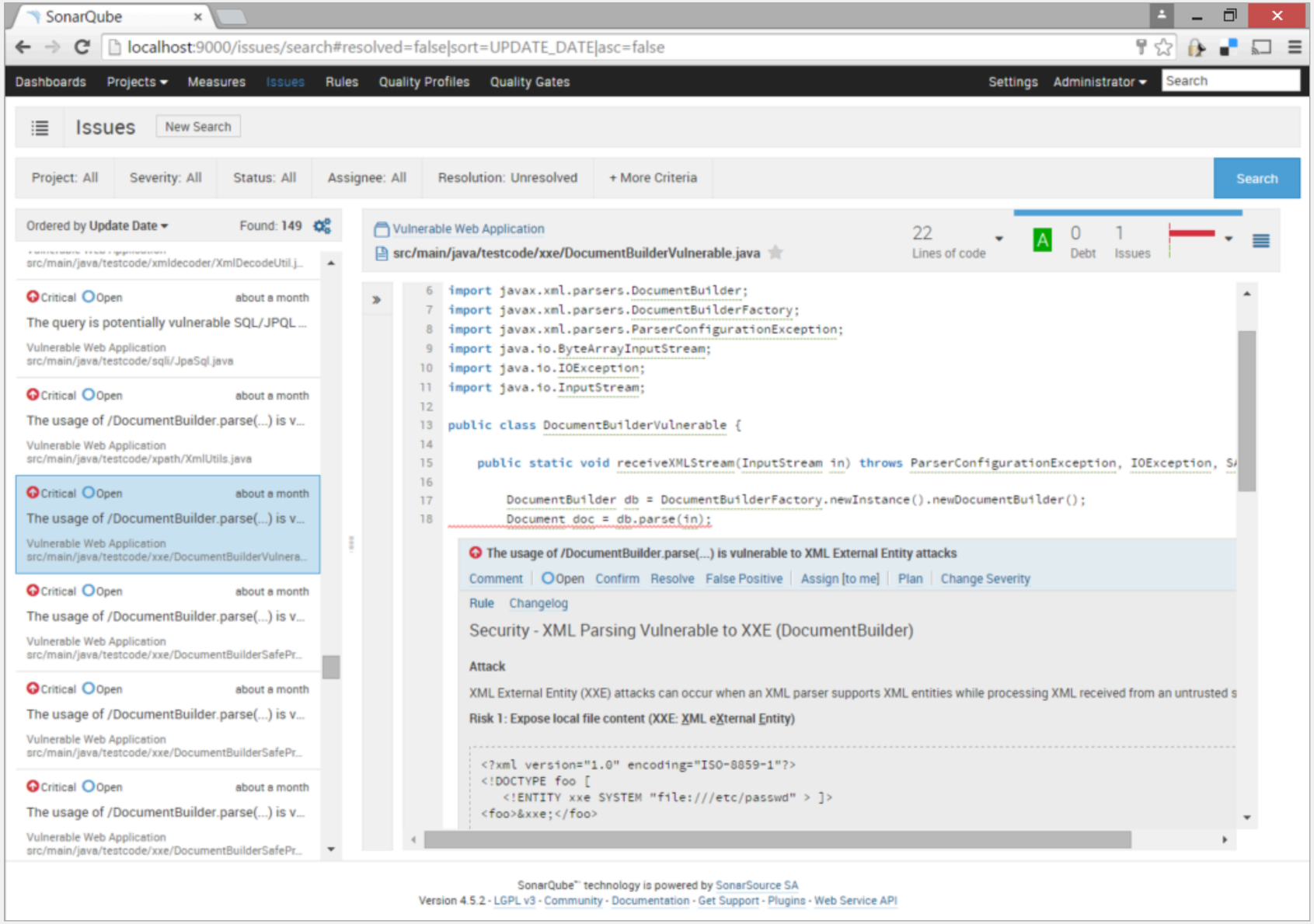
By using the dependency-check tool, to help R&D team to effectively find out the 3rd frameworks or libraries with the high risk security issues.

Development tool verification

In order to avoid the similar security issue caused by Apple's vulnerable developer tool XcodeGhost, the development tools used by the team should be verified to be secure.

05. Testing and Release

05. 实践 - 静态代码安全扫描



05. 实践 - CI/CD 中集成安全扫描工具

配置 >> 运行 >> 获取报告

```
apply plugin: "java"
apply plugin: "idea"

apply plugin: "security-zap"

buildscript {
    repositories {
        mavenCentral()
    }
    dependencies {
        classpath(
                'com.thoughtworks.tools:security-zap:1.0.5'
        )
    }
}
```

```
➜  customer-api git:(master) ✗ gradle zapStart build -Dzap.proxy=localhost:7070 zapReport
:zapStop
stopping zap
Warning: failed to stop ZAP due to connection refused or ZAP already stopped.
:zapStart
Starting ZAP [apikey: 2LZ77g9YVi]
waiting ZAP
waiting ZAP
waiting ZAP
ZAP started
exclusion rules are removed
urls match following regex will be excluded:
.*/css/.*
.*/js/.*
.*/fonts/.*
.*\.css
.*\.js
```

2. Security Alerts Summary

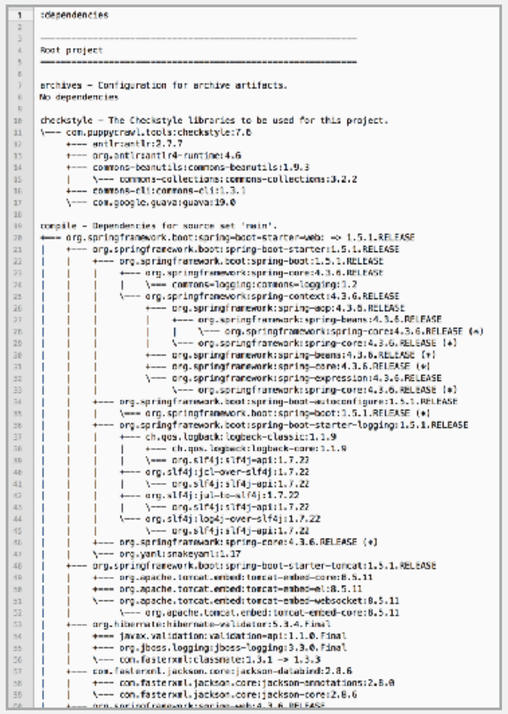
Number of alerts in total: 541

Alerts by severity	Amount
High	2
Medium	5
Low	360
Informational	174

3. Security Alerts By Classification

Classification	Amount
Cross Site Scripting (Reflected)	1
SQL Injection	1
Session ID in URL rewrite	4

05. 实践 -第三方依赖安全检查



```
:dependencies

------------------------------------------------------------
Root project
------------------------------------------------------------

archives - Configuration for archive artifacts.
No dependencies

checkstyle - The Checkstyle libraries to be used for this project.
\--- com.puppycrawl.tools:checkstyle:7.6
     +--- antlr:antlr:2.7.7
     +--- org.antlr:antlr4-runtime:4.6
     +--- commons-beanutils:commons-beanutils:1.9.3
     |    \--- commons-collections:commons-collections:3.2.2
     +--- commons-cli:commons-cli:1.3.1
     \--- com.google.guava:guava:19.0

compile - Dependencies for source set 'main'.
+--- org.springframework.boot:spring-boot-starter-web -> 1.5.1.RELEASE
|    +--- org.springframework.boot:spring-boot-starter:1.5.1.RELEASE
|    |    +--- org.springframework.boot:spring-boot:1.5.1.RELEASE
|    |    |    +--- org.springframework:spring-core:4.3.6.RELEASE
|    |    |    |    \--- commons-logging:commons-logging:1.2
|    |    |    \--- org.springframework:spring-context:4.3.6.RELEASE
|    |    |         +--- org.springframework:spring-aop:4.3.6.RELEASE
|    |    |         |    +--- org.springframework:spring-beans:4.3.6.RELEASE
|    |    |         |    |    \--- org.springframework:spring-core:4.3.6.RELEASE (*)
|    |    |         |    \--- org.springframework:spring-core:4.3.6.RELEASE (*)
|    |    |         +--- org.springframework:spring-beans:4.3.6.RELEASE (*)
|    |    |         +--- org.springframework:spring-core:4.3.6.RELEASE (*)
|    |    |         \--- org.springframework:spring-expression:4.3.6.RELEASE
|    |    |              \--- org.springframework:spring-core:4.3.6.RELEASE (*)
|    |    +--- org.springframework.boot:spring-boot-autoconfigure:1.5.1.RELEASE
|    |    |    \--- org.springframework.boot:spring-boot:1.5.1.RELEASE (*)
|    |    +--- org.springframework.boot:spring-boot-starter-logging:1.5.1.RELEASE
|    |    |    +--- ch.qos.logback:logback-classic:1.1.9
|    |    |    |    +--- ch.qos.logback:logback-core:1.1.9
|    |    |    |    \--- org.slf4j:slf4j-api:1.7.22
|    |    |    +--- org.slf4j:jcl-over-slf4j:1.7.22
|    |    |    |    \--- org.slf4j:slf4j-api:1.7.22
|    |    |    +--- org.slf4j:jul-to-slf4j:1.7.22
|    |    |    |    \--- org.slf4j:slf4j-api:1.7.22
|    |    |    \--- org.slf4j:log4j-over-slf4j:1.7.22
|    |    |         \--- org.slf4j:slf4j-api:1.7.22
|    |    +--- org.springframework:spring-core:4.3.6.RELEASE (*)
|    |    \--- org.yaml:snakeyaml:1.17
|    +--- org.springframework.boot:spring-boot-starter-tomcat:1.5.1.RELEASE
|    |    +--- org.apache.tomcat.embed:tomcat-embed-core:8.5.11
|    |    +--- org.apache.tomcat.embed:tomcat-embed-el:8.5.11
|    |    \--- org.apache.tomcat.embed:tomcat-embed-websocket:8.5.11
|    |         \--- org.apache.tomcat.embed:tomcat-embed-core:8.5.11
|    +--- org.hibernate:hibernate-validator:5.3.4.Final
|    |    +--- javax.validation:validation-api:1.1.0.Final
|    |    +--- org.jboss.logging:jboss-logging:3.3.0.Final
|    |    \--- com.fasterxml:classmate:1.3.1 -> 1.3.3
|    +--- com.fasterxml.jackson.core:jackson-databind:2.8.6
|    |    +--- com.fasterxml.jackson.core:jackson-annotations:2.8.0
|    |    \--- com.fasterxml.jackson.core:jackson-core:2.8.6
|    +--- org.springframework:spring-web:4.3.6.RELEASE
```

以前

通过媒体被动获取漏洞信息

人工审查

耗时长

&

不可持续

现在

全自动化的工具

迅速获取安全质量

&

持续监控

DEMO ONLINE

05. 实践 - 自动化安全测试用例

基于安全需求制定安全测试用例

Given an anonymous visitor
When I try to access report page without authentication
Then I was been redirected to login page

Given a user without report access permission
When I try to access report page with authentication
Then I was been redirected to error page

Given a system manager
When I try to access report page with authentication
Then I can access report page successfully

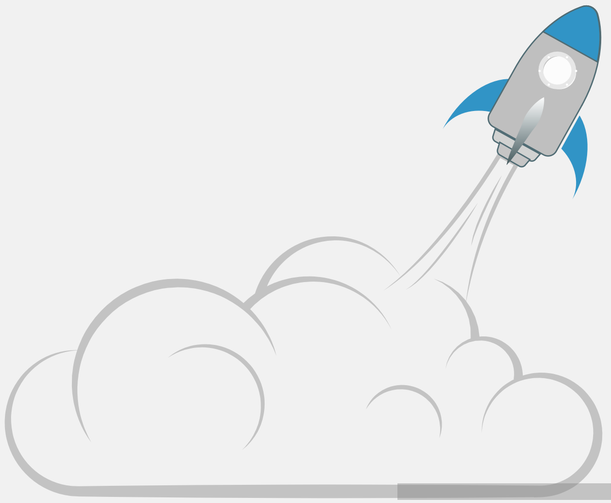
·将安全测试用例通过普通的测试来实现

```
public void anonymousVisitorCanNotAccessReportPage() {
    Page currentPage = accessReportPage();
    assertThat(currentPage, is(LOGIN_PAGE));
}
```

```
public void userWithoutProperPermissionCanNotAccessReportPage() {
    loginAsMember();
    Page currentPage = accessReportPage();
    assertThat(currentPage, is(PERMISSION_REQUIRED_ERROR_PAGE));
}
```

```
public void managerCanAccessReportPage() {
    loginAsManager();
    Page currentPage = accessReportPage();
    assertThat(currentPage, is(REPORT_PAGE));
}
```

06. Operation and Maintenance



Your system is secure currently, but it doesn't mean it is secure in the future, the reason is we just address all KNOWN issues.The unknown security issues will be raised as the time goes on, so regular security scan can help the team to find out and fix the new security issued in the system at a very early stage.

Regular Security Scan

Follow the industry security events, and take appropriate actions for the issues which have big impact on the system to reduce security risks.

Industry Security Event Monitoring

ThoughtWorks® | IT大咖说 知识分享平台

THANK YOU