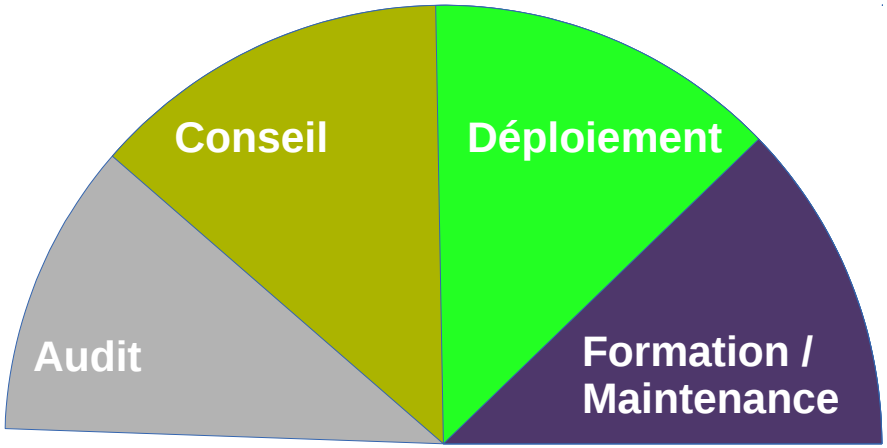
La Sécurité Applicative en DevSecOps avec JAZZ et APPSCAN

ABlogiX – Présentation

- Présentation
  - Nos activités
  - La Sécurité Applicative
  -

Nos activités

- ABlogiX est une société fondée en 2007 par Arnaud Bourlier
- 2 axes de compétences : **Qualité & Sécurité**
  - **Amélioration** des **processus** de « **développement logiciel et système** »
  - **Amélioration** de la **Sécurité du Système d'Information**



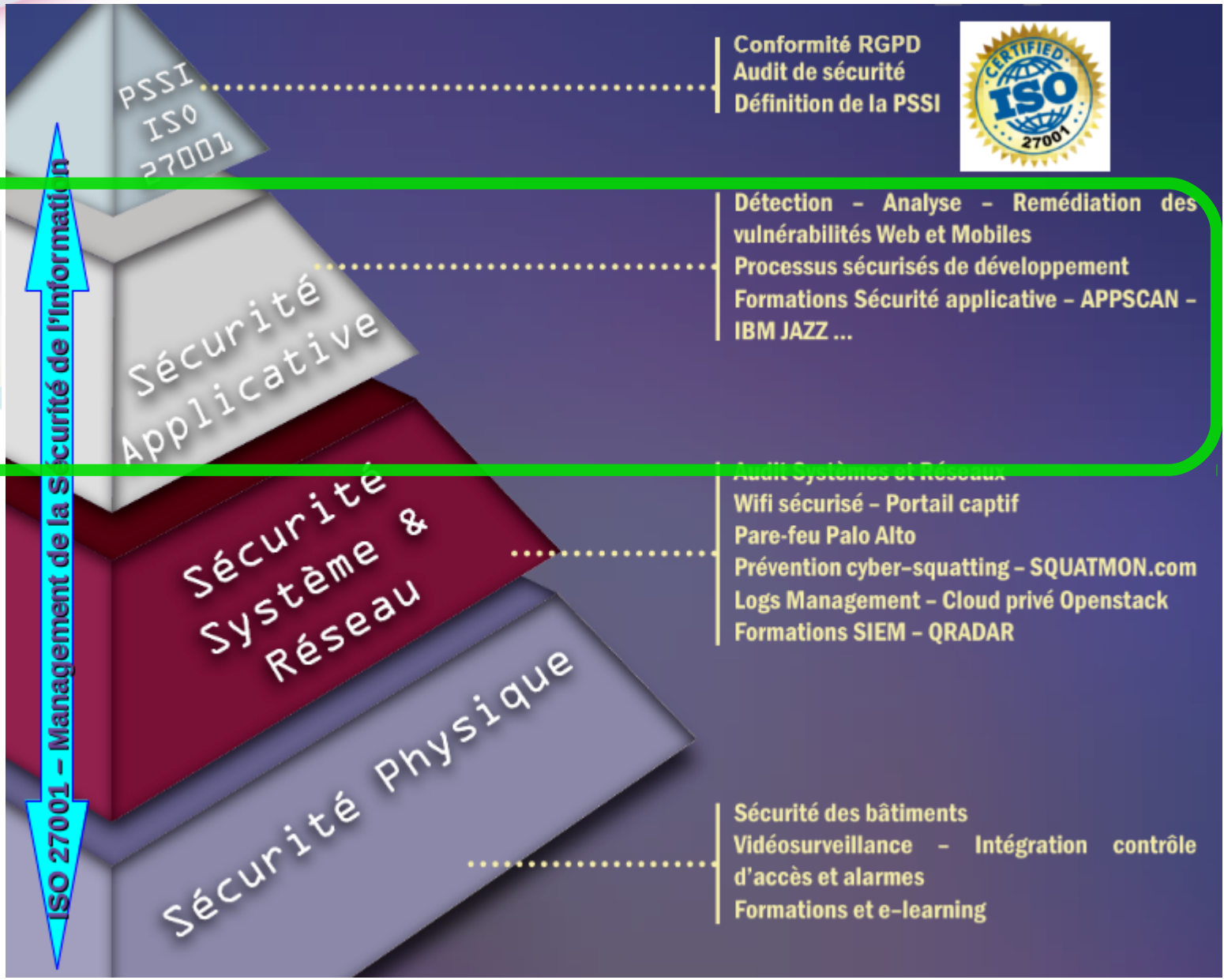
La Sécurité Applicative dans la pyramide de la SSI

Parmi les produits déployés :

AppScan
IBM. Rational.
Dynamic & Static Application Vulnerability Testing

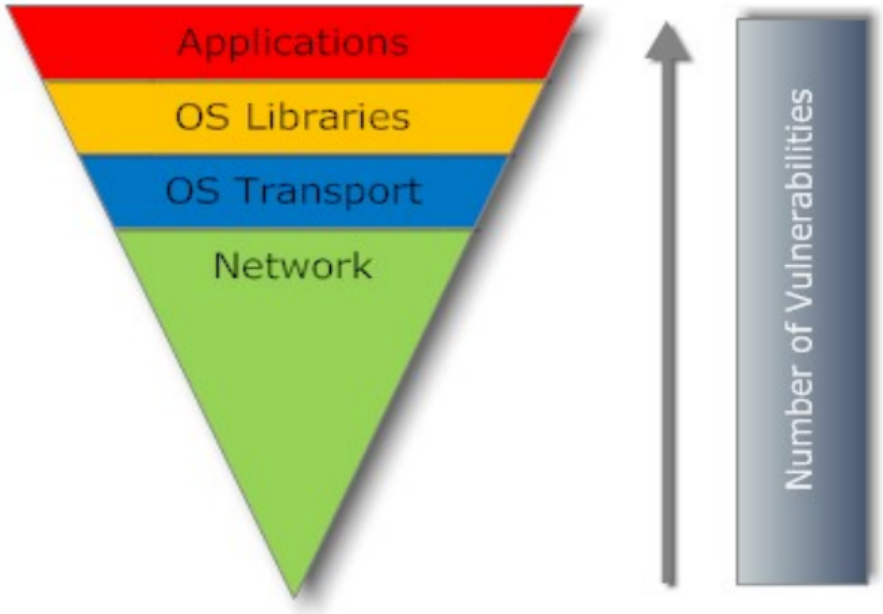
openstack™

QRadar
IBM.



Pourquoi les applications sont-elles si vulnérables ?

- Les développeurs se focalisent sur l'**aspect fonctionnel** pour respecter les **délais et budget** – mais pas nécessairement sur la **sécurité**
- Les **développeurs** ne sont généralement **pas formés aux pratiques de code sécurisé**
- **43 % des failles concernent les applications web** alors que la grande majorité des investissements en sécurité des entreprises concernent l'infrastructure et le réseau.



Comment améliorer la sécurité des applications

Différentes démarches complémentaires :

- **Audit** de sécurité de l'architecture logicielle (i.e. respect et robustesse des 3 « A »)
- **Scan** des vulnérabilités en mode statique et dynamique (Revue/Pentest/Automatique)
- **Déploiement** de Scanner DAST et SAST et intégration dans le Cycle de Développement

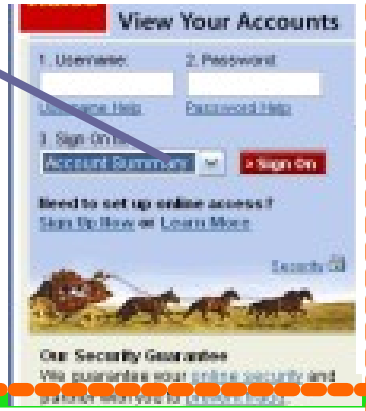
- **Formation** des développeurs (France & International) : *200 étudiants*
  - ✔ Sécurité des applications – Identification/remédiation des vulnérabilités

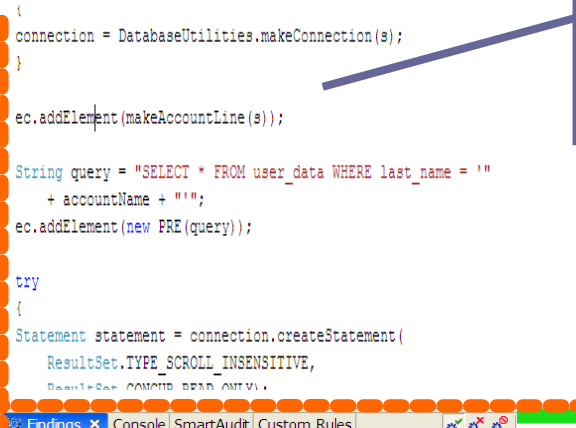
Les techniques de recherches des vulnérabilités : manuel/ Automatique

4 techniques complémentaires :

Manuel

Tests d'intrusions manuels

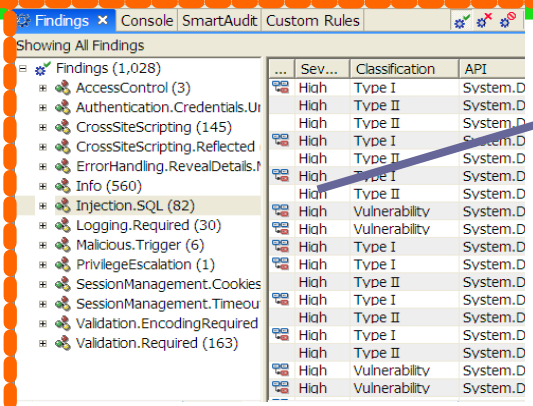




```
connection = DatabaseUtilities.makeConnection(s);
}

ec.addElement(makeAccountLine(s));

String query = "SELECT * FROM user_data WHERE last_name = '"
    + accountName + "'";
ec.addElement(new PRE(query));

try
{
Statement statement = connection.createStatement(
    ResultSet.TYPE_SCROLL_INSENSITIVE,
```

Audit et Revue Manuelle du Code Source

Exécution — **Code**



Analyse Statique Automatisée

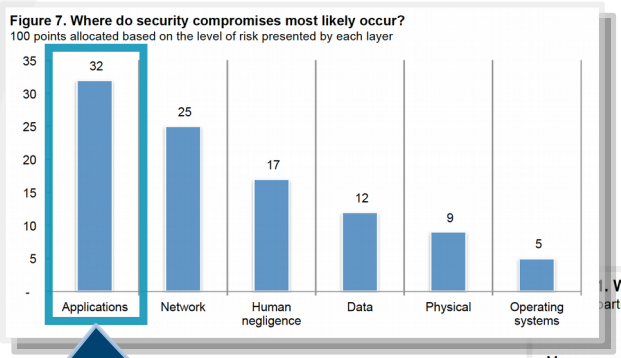
Scan Automatique d'application

Automatique

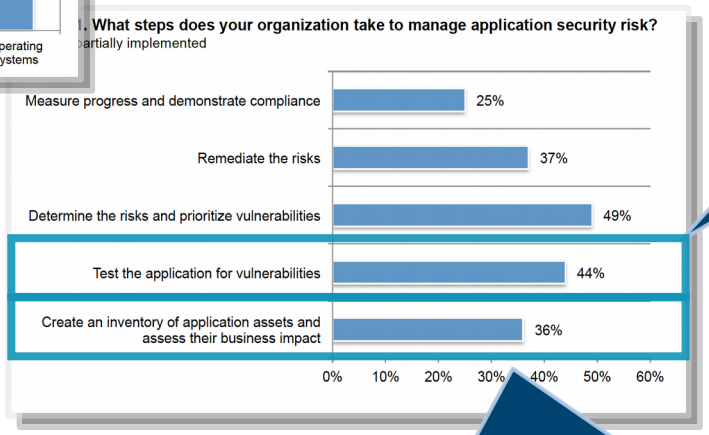
Etat de la situation sur la sécurité des applications

Seulement 11% des entreprises ont des programmes de sécurité applicatives complets

- Etude indépendante menée par Ponemon Institute LLC - mars 2016



Les applications sont les plus vulnérables aux menaces



Que dire des 56% restants

64% ignorent le nombre d'applications dans leur société

Ponemon INSTITUTE

Full Survey Results

Trouver les erreurs le plus tôt possible reste essentiel

Durant le développement

$80 / défaut

Durant la phase de Build

$240 / défaut

Durant les phases de test

$960 / défaut

Durant la phase de production

$7,600 / défaut

80% des coûts de développement sont consacrés à l'identification et à la correction des défauts !

Source: National Institute of Standards and Technology

- Coût d'une violation de données 7,2 M $
- 80 jours pour la détecter
- Plus de quatre mois (123 jours) à résoudre

Source: Ponemon Institute

Trouver les erreurs le plus tôt possible reste essentiel

3 facteurs de réussite essentiels: automatisation, traçabilité et complétude

Application Security Management

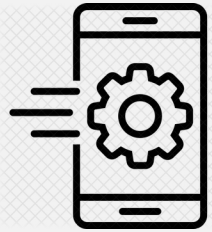
Utilisez une console unique pour gérer les risques liés aux applications, les résultats des tests et les rapports

Dynamic

Identifier les vulnérabilités dans les applications en cours d'exécution

Static

Analyse de trace du code d'application



Mobile

IAS test pour les binaires Mobile

Open Source

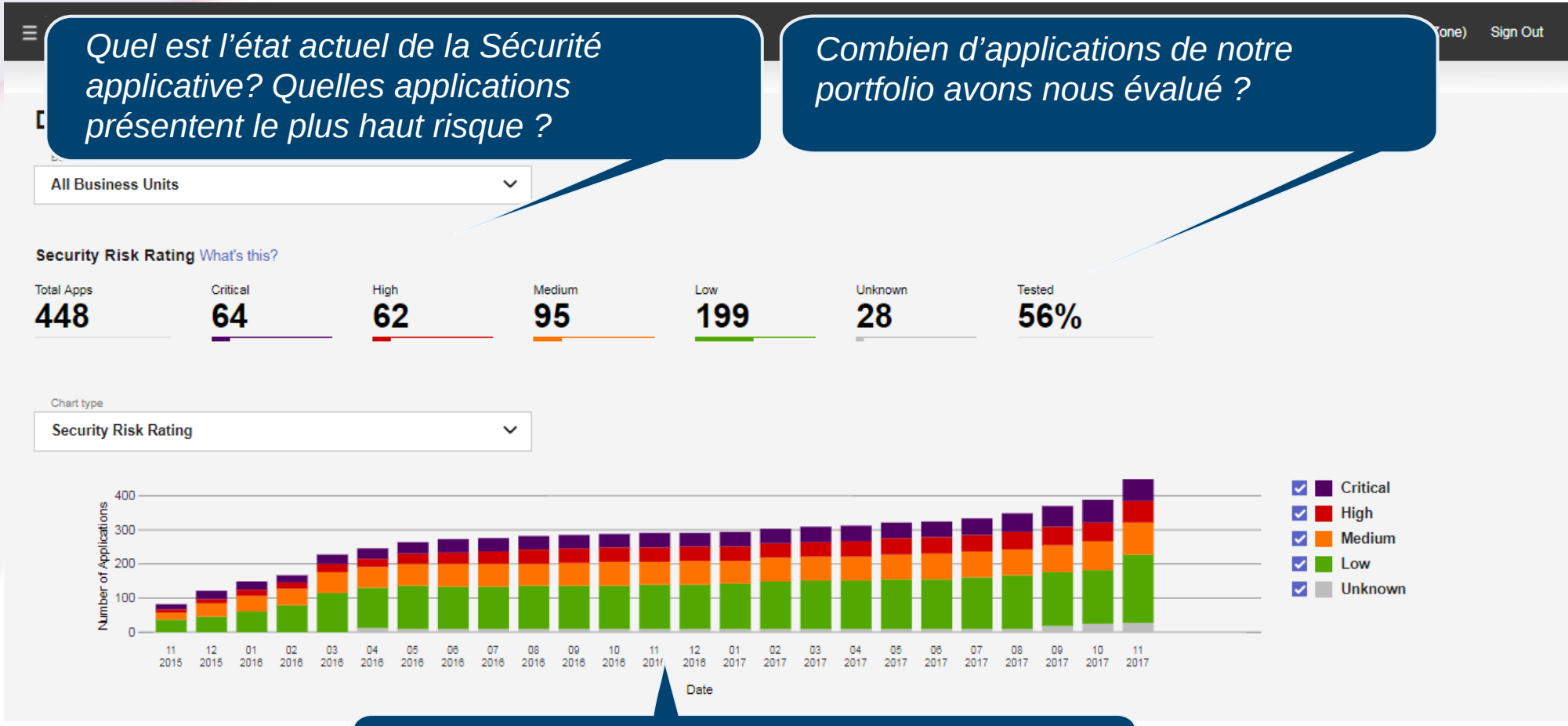
Identifier les vulnérabilités introduites par les packages open source

APPSCAN on-Premise ou **APPSCAN on-Cloud (ASoC)**

APPSCAN : Un Dashboard unique pour l'Application Security Management

Quel est l'état actuel de la Sécurité applicative? Quelles applications présentent le plus haut risque ?

Combien d'applications de notre portfolio avons nous évalué ?



L'état de notre Sécurité Applicative s'améliore-t-elle?

Dasboard principal de la Solution IBM ASoc

Gérez les risques de votre organisation

Sommes-nous en train de réduire les risques ?

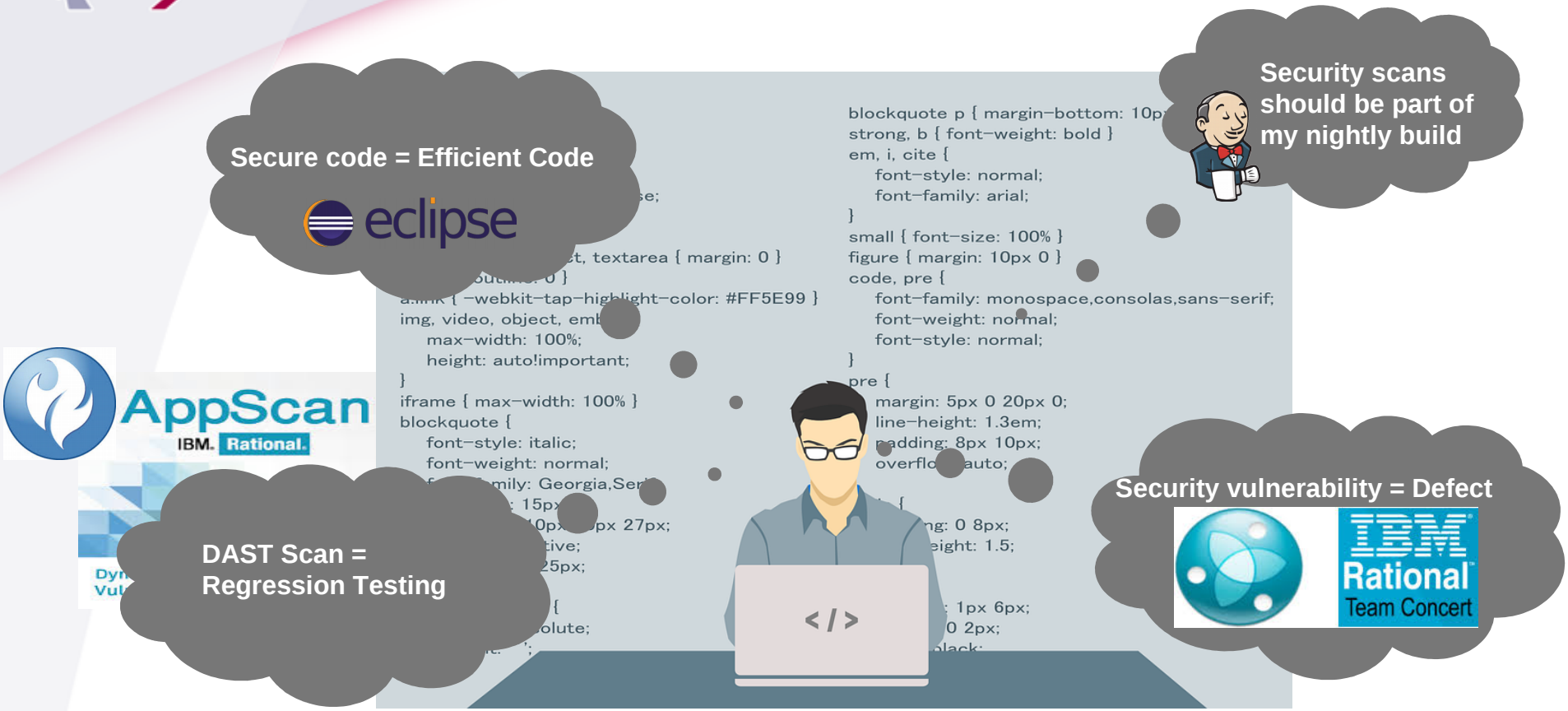
Quel est le statut actuel ?

Quel est le risque le plus répandu ?

Les applications répertoriées

Évolution des risques applicatifs ?

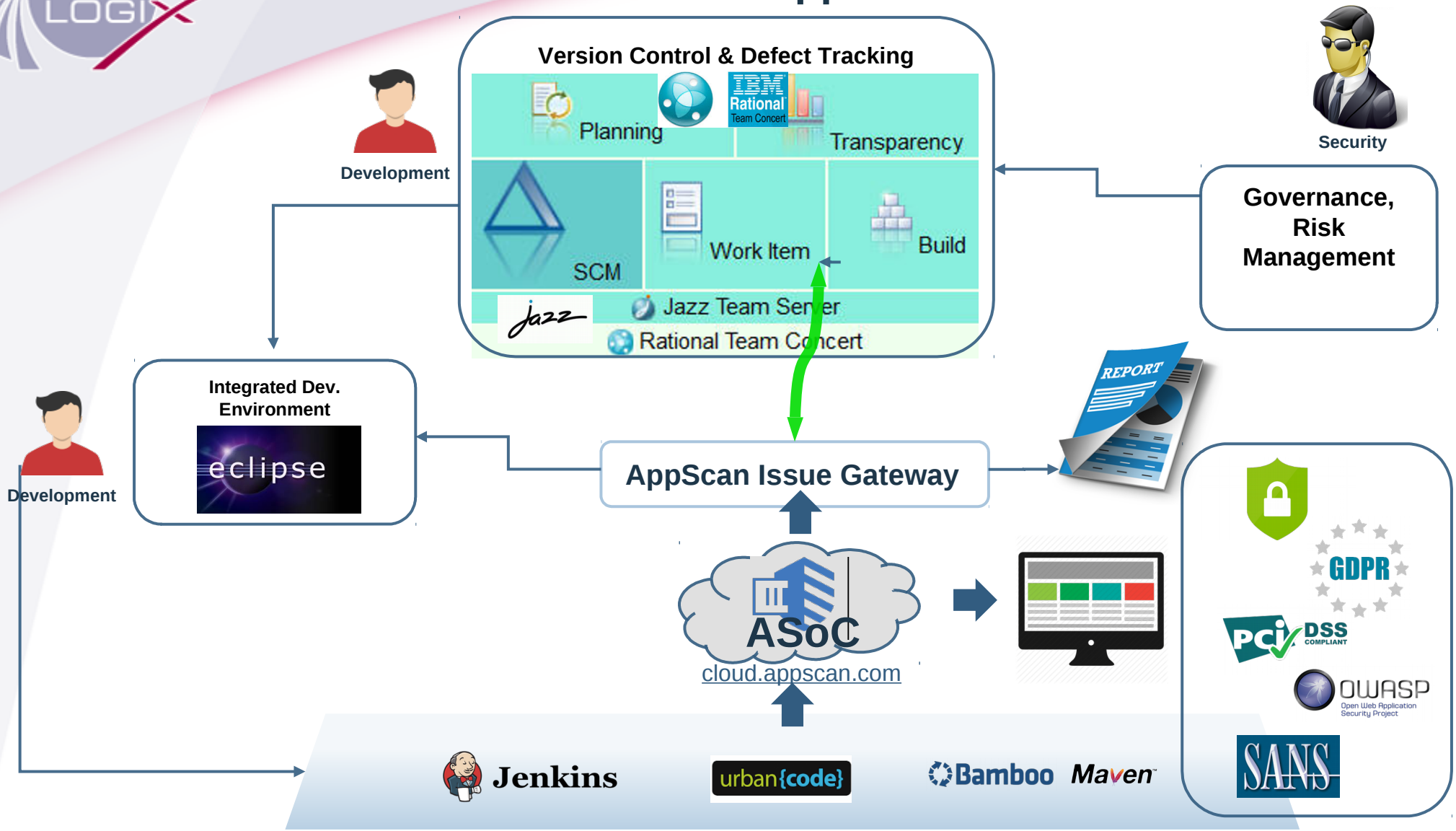
Intégration dans les outils de développement



https://www.ibm.com/support/knowledgecenter/fr/SSYJJF_1.0.0/ApplicationSecurityonCloud/appseccloud_rest_apis.html

L'interface d'API REST intégrée vous permet de visualiser les services Web RESTful. La documentation API est élaborée à l'aide de Swagger. Vous pouvez y tester des opérations API et consulter instantanément les résultats afin d'examiner vos applications plus rapidement.

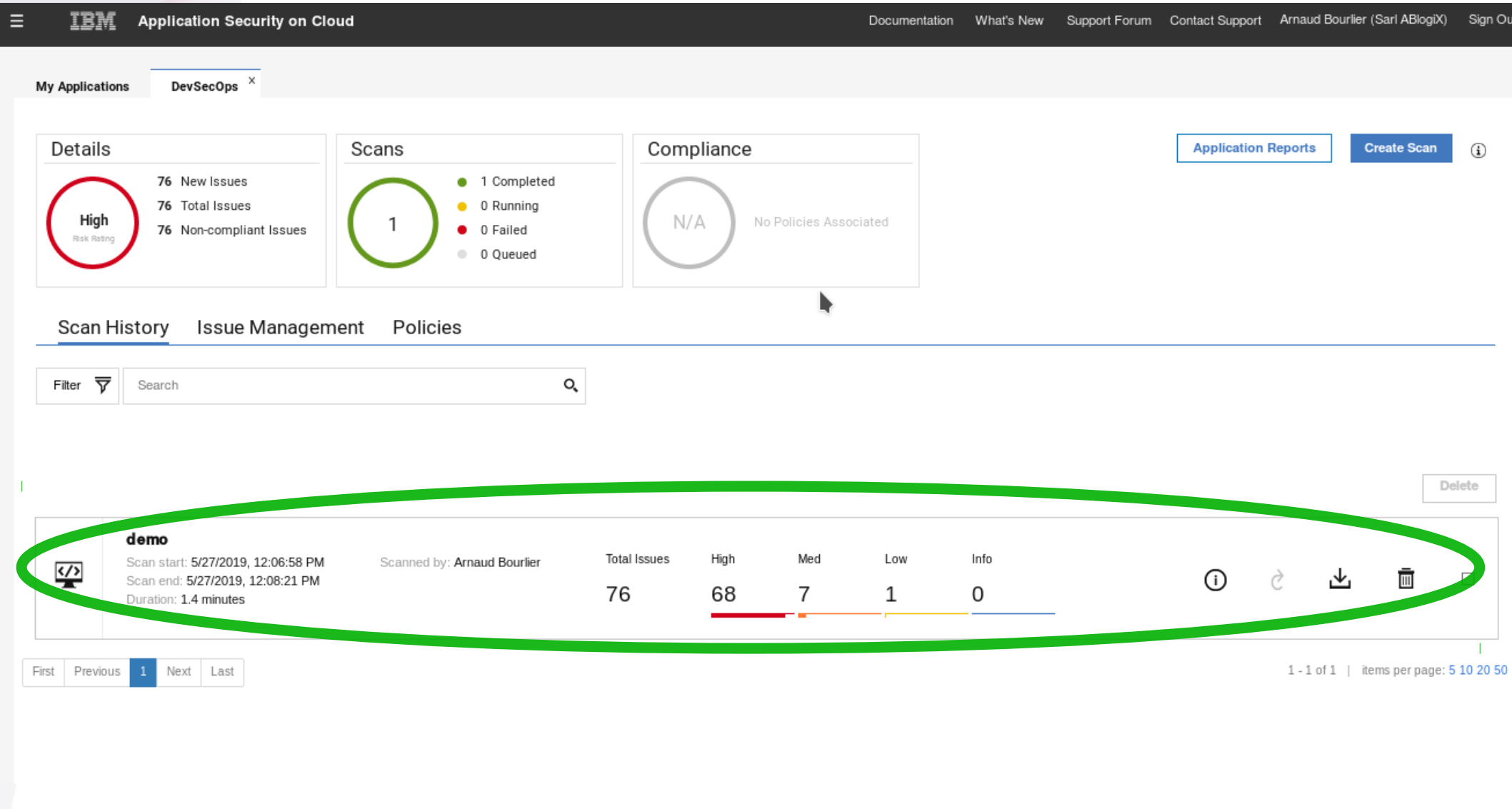
DevSecOps : Intégration de ASoC dans le Cycle de Développement



- **ASoC** : AppScan on Cloud is available at : https://cloud.appscan.com
- https://jazz.net met à disposition l'outil collaboratif JAZZ RTC de Gestion de Version & de Gestion des Changements.
- APPSCAN.Issue.Gateway : disponible sur https://github.com/hclproducts/appscan-issue-gateway permet de fournir l'intégration entre JAZZ RTC et ASoC

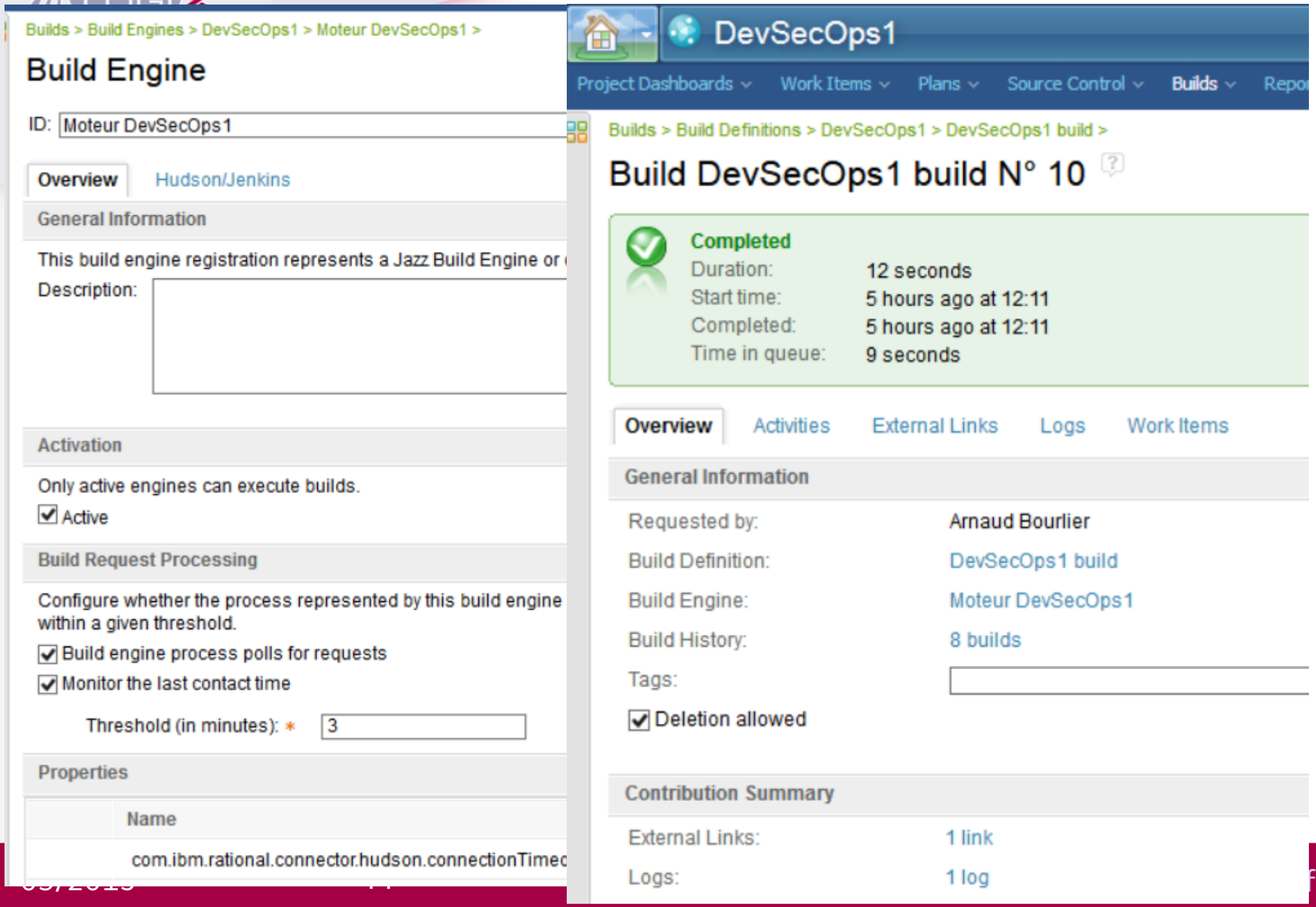
ASoC – Intégration avec JAZZ RTC

- Création d'un projet « demo »
- Exécution du scan



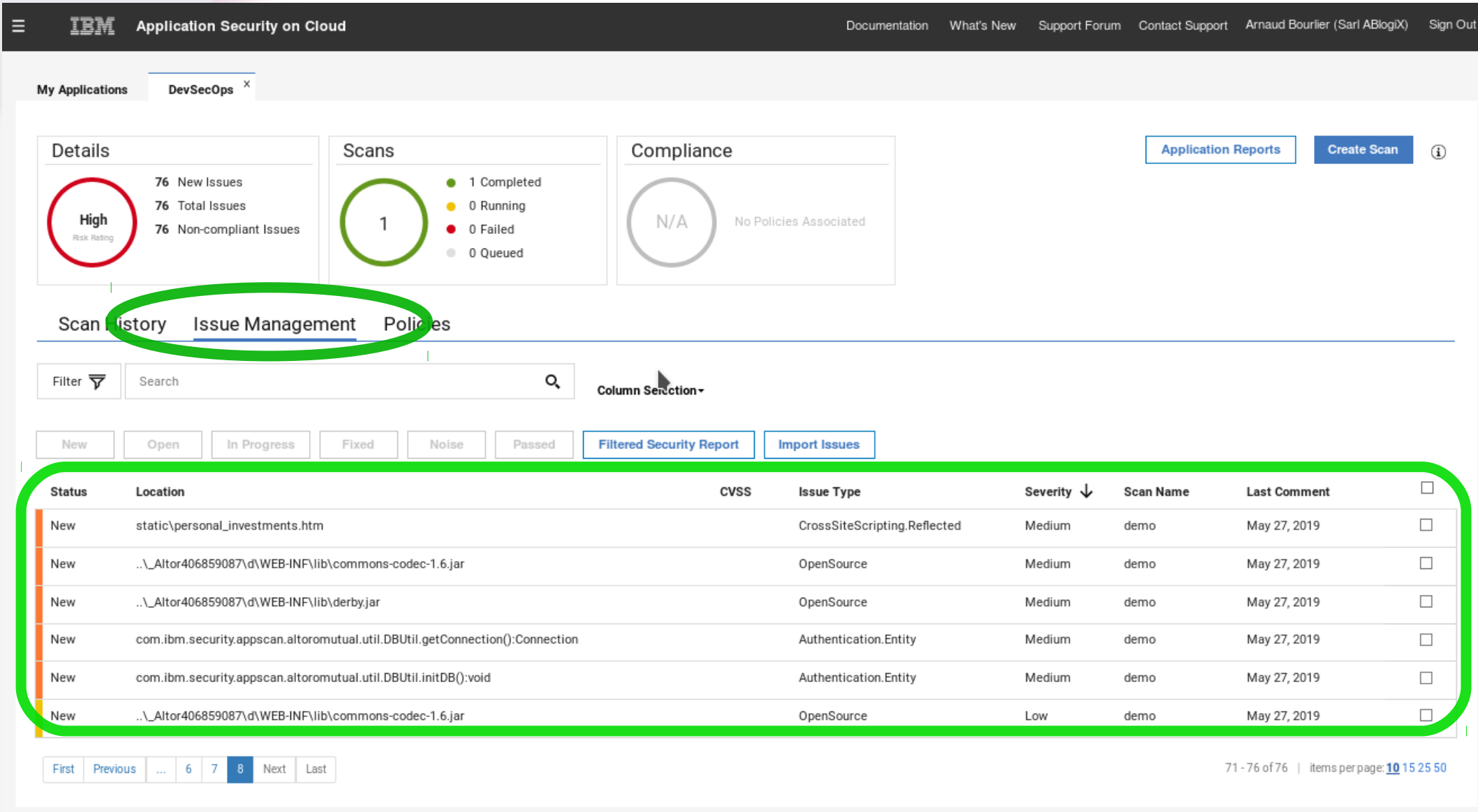
ASoC – Intégration avec JAZZ RTC

- Le Build RTC est déclenché manuellement ou sur évènement



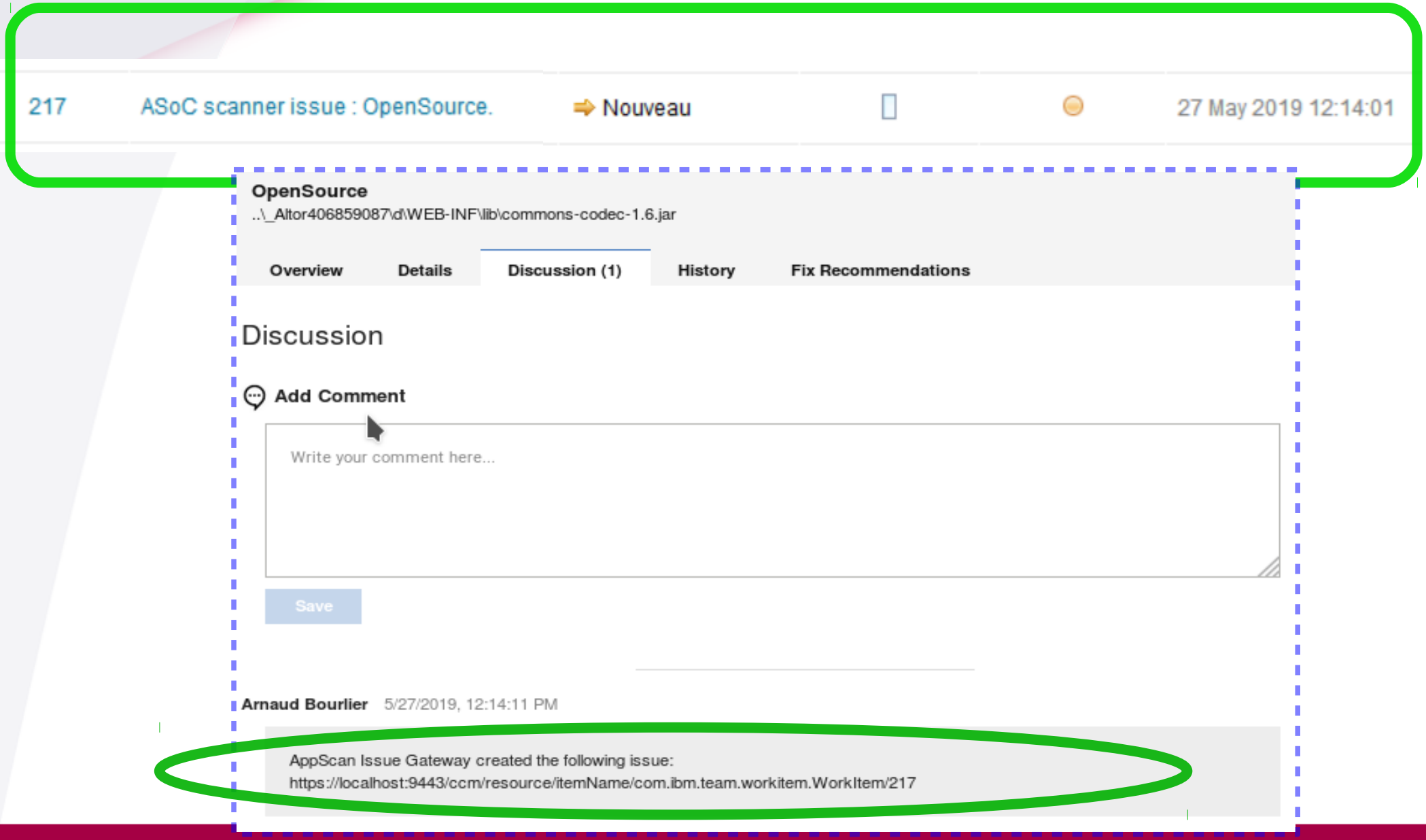
ASoC – Intégration avec JAZZ RTC

- les vulnérabilités identifiées sont détaillées sous l'onglet « Issue Management »
- RTC déclenche la création des Issues vers des Work Items



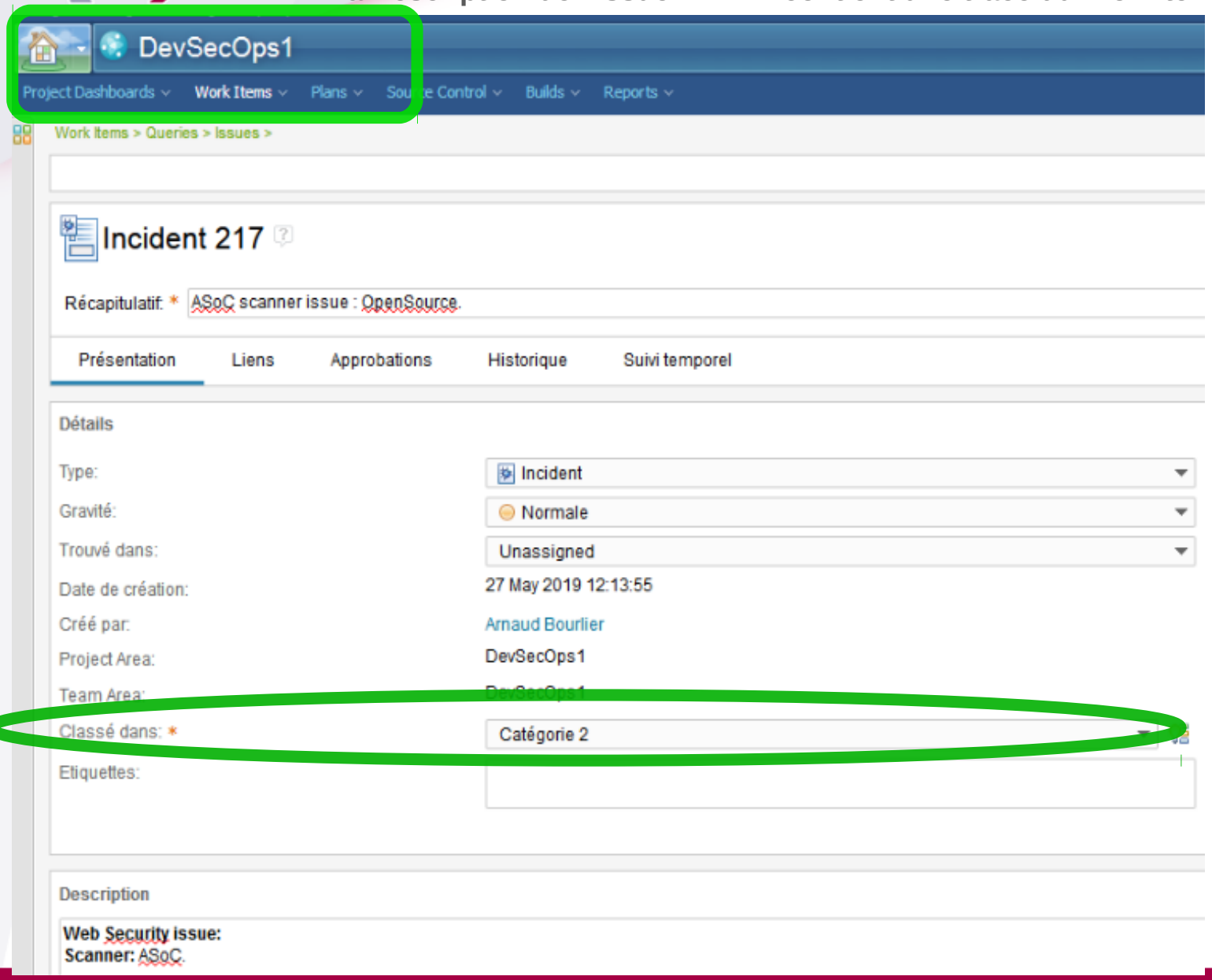
ASoC – Intégration avec JAZZ RTC

- L'Issue « Open Source » a généré un WorkItem « 217 »
- La Description de l'Issue contient une trace du WorkItem créé « 217 »



ASoC – Intégration avec JAZZ RTC

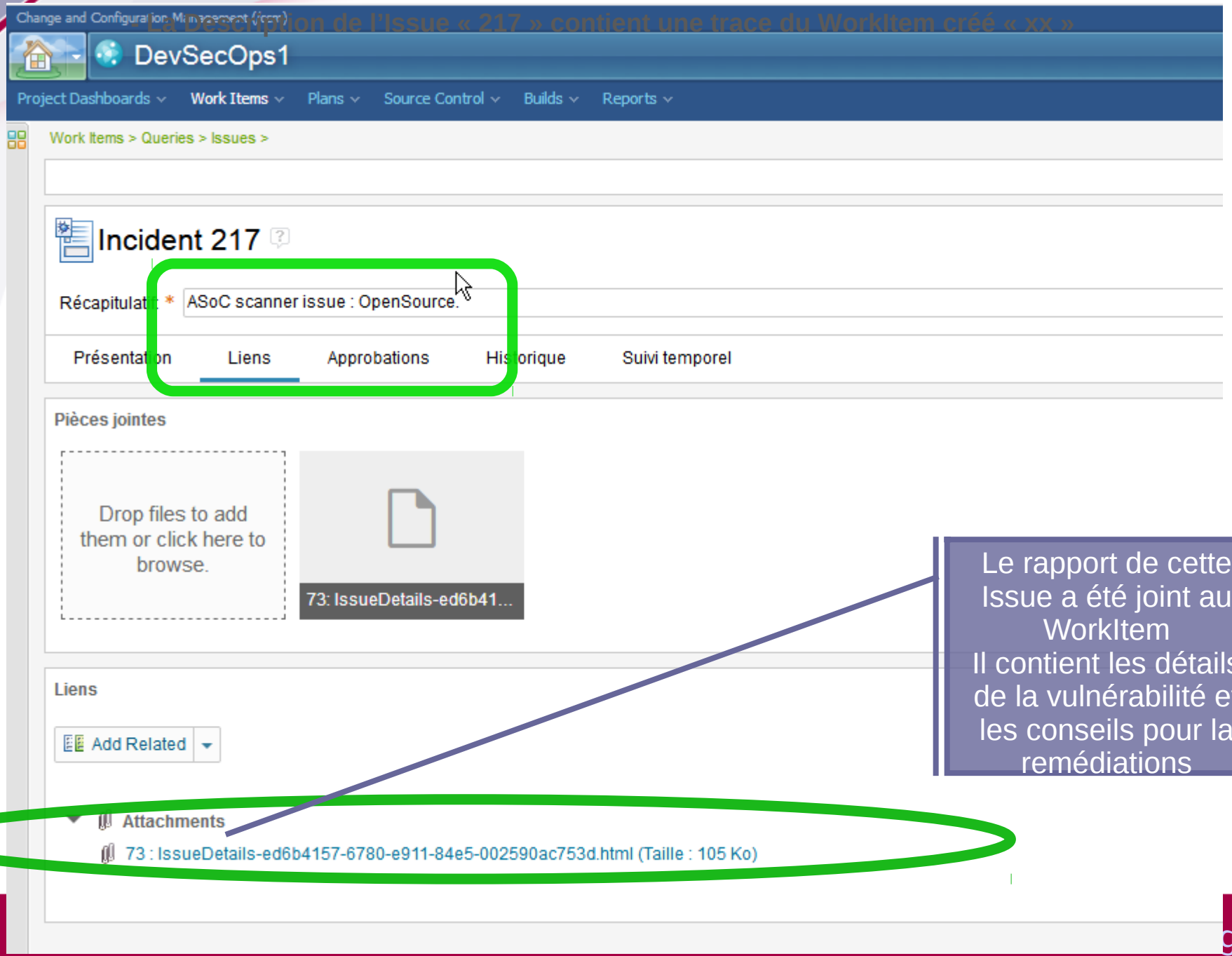
- Le WorkItem « 217 » a été créé sous la Project Area RTC
- La Description de l'Issue « 217 » contient une trace du WorkItem créé « xx »



ASoC – Intégration avec JAZZ RTC

- Le WorkItem « 217 » a été créé sous la Project Area RTC

- La Description de l'Issue « 217 » contient une trace du WorkItem créé « xx »



Le rapport de cette Issue a été joint au WorkItem
Il contient les détails de la vulnérabilité et les conseils pour la remédiations

ASoC – Intégration avec JAZZ RTC

- Le rapport de cette vulnérabilité donne les détails

IBM A

Appli

Name: Sing

Security Report for
Business Impact: H
Created: Monday, M
Contains: 1 issues (

Summ

Total security is

Issue Types: 1

OpenSource

Issues

L **SAST: OpenSource 1**

Issue 1 of 1

Issue ID:	-557898642
Severity:	Low
Status	New
Classification	Definitive
Location	.._Altor406859087\d\WEB-INF\lib\commons-codec-1.6.jar
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Monday, May 27, 2019
Last Updated	Monday, May 27, 2019
File:	.._Altor406859087\d\WEB-INF\lib\commons-codec-1.6.jar
Name:	2009-0001
Description:	Not all "business" method implementations of public API in Apache Commons Codec 1.x are thread safe, which might disclose the wrong data or allow an attacker to change non-private fields. Updated 2018-10-07 - an additional review by WhiteSource research team could not indicate on a clear security vulnerability
URL:	https://issues.apache.org/jira/browse/CODEC-55

Issue 1 of 1 - Details

Issue 1 of 1 - Audit Trail

05/27/2019 10:07:24	IssueTypeName: → OpenSource Status: → New Location: → .._Altor406859087\d\WEB-INF\lib\commons-codec-1.6.jar Severity: → Low Scanner: → AppScan Static Analyzer

ASoC – utilisation des capacités cognitives pour les Tests de Sécurité Applicative

Application Security Advisor

- **Intelligent Code Analytics (ICA)**

  Étend la couverture des analyses et élimine les **Faux Négatifs** en générant des Politiques de Sécurité pour **TOUT** framework utilisé par une application durant l'analyse des traces.

- **Intelligent Finding Analytics (IFA)**

  Réduit les **Faux Positifs** **jusqu'à** 98% & élimine les longs processus de revues des rapports en fournissant des revues automatisées complètes des anomalies des tests de Sécurité Applicative.

- **Simple Fix Group recommendations**

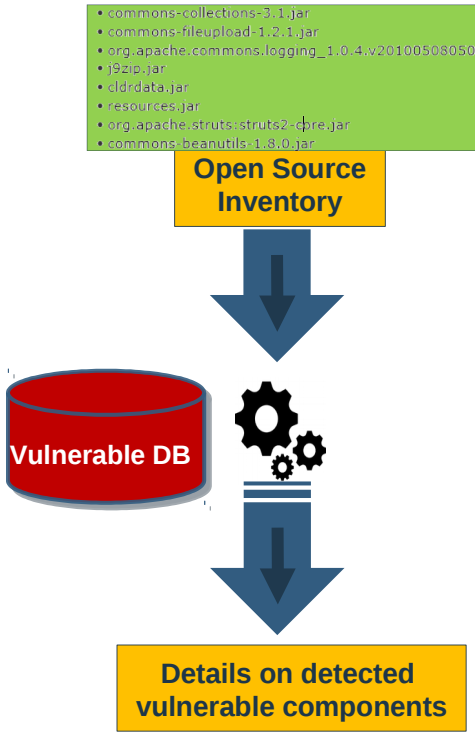
  Fournit des recommandations de correctifs qui aident l'équipe de développement à résoudre de multiples vulnérabilités avec un seul correctif.

ASoC – L'analyse Applicative en 3 phases

Etape 1: Analyse Open Source

L'outil d'analyse open source construit une liste de tous les composants de l'application et vérifie si l'un de ceux-ci présente des vulnérabilités.

- Configuration minimale requise pour effectuer une analyse.
- Les analyses sont extrêmement rapides et claires
  - *Environ 80% à 90% des défauts du code des applications actuelles proviennent des composants open source.
- IBM ASoc trouve rapidement les sites vulnérables avec Google Dork ou des outils conçus par les pirates pour détecter les nouveaux CVE publiés.



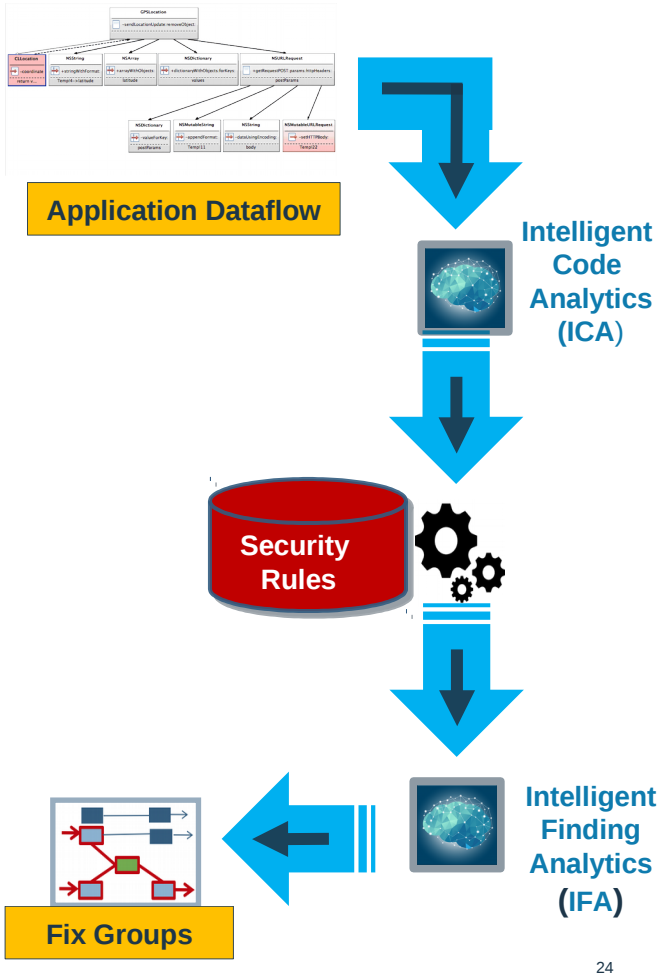
* Forrester: *How To Leverage DevOps Trends To Strengthen Applications Dec. 2016*

ASoC – L'analyse Applicative en 3 phases

Etape 2: Analyse Statique

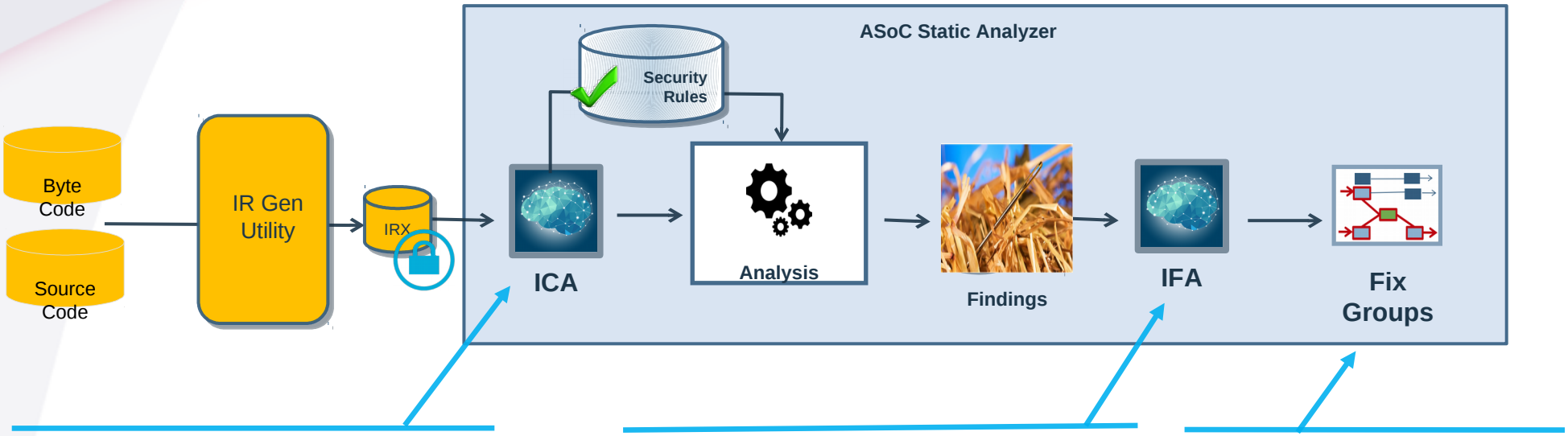
L'**analyse statique** a pour but de rechercher les vulnérabilités dans le code de l'application elle-même. L'une des méthodes consiste à détecter les flux de données qui ne sont pas désinfectés.

- Une petite configuration est nécessaire pour effectuer une analyse.
- L'analyse de petites applications / micro-services s'effectue en moins de 2 minutes.
- Les vulnérabilités signalées orientent généralement le développeur vers le code à corriger via l'interface de ses outils de développement.
- Des résultats très précis grâce à l'intelligence cognitive et au machine learning.



ASoC – L'analyse Applicative en 3 phases

Les fonctionnalités cognitives pour le Static Analyzer



Intelligent Code Analytics (ICA)

Élimine les Faux Négatifs
- couvre les mises-à-jour des Langages
- couvre tous les Frameworks

Intelligent Finding Analytics (IFA)

Élimine les Faux Positifs & anomalies in-intéressantes
- A donné un gain jusqu'à 98.91% sur un cas d'usage client

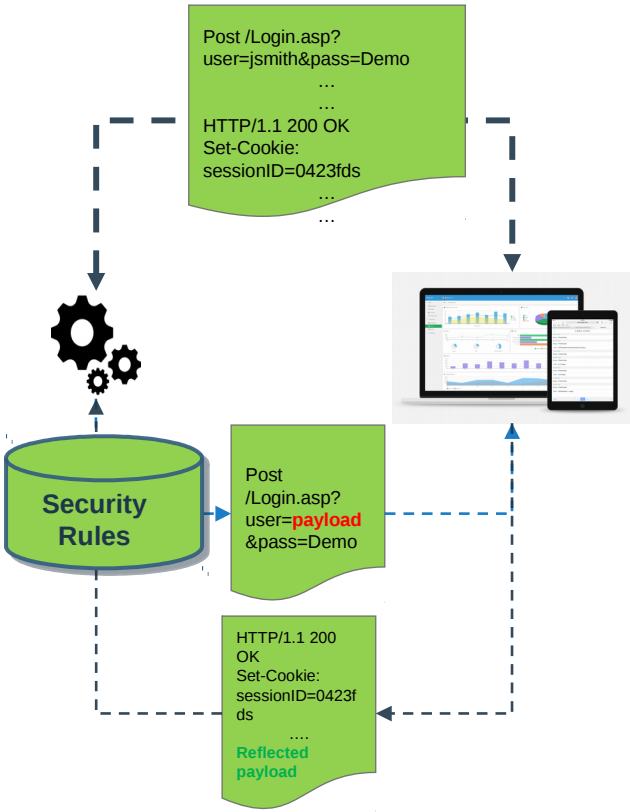
Identifie les endroits optimaux dans le code source pour corriger de multiple Vulnérabilités

ASoC – L'analyse Applicative en 3 phases

Etape 3: Analyse Dynamique

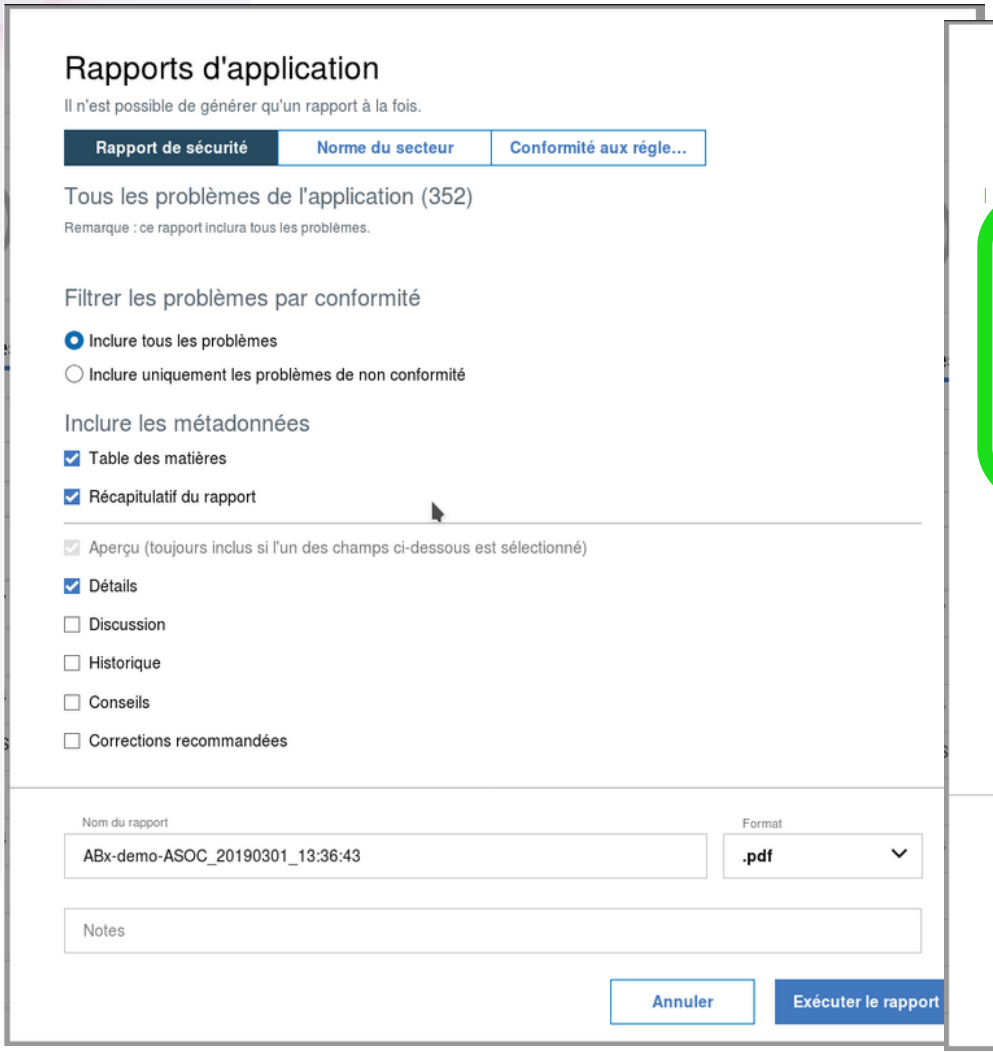
L'**analyse dynamique** vise à identifier des vulnérabilités dans une application en cours d'exécution.
La 1ère étape consiste à collecter des informations sur l'application
La 2nd consiste à envoyer des requêtes HTTP modifiées à l'application et en examiner les réponses.

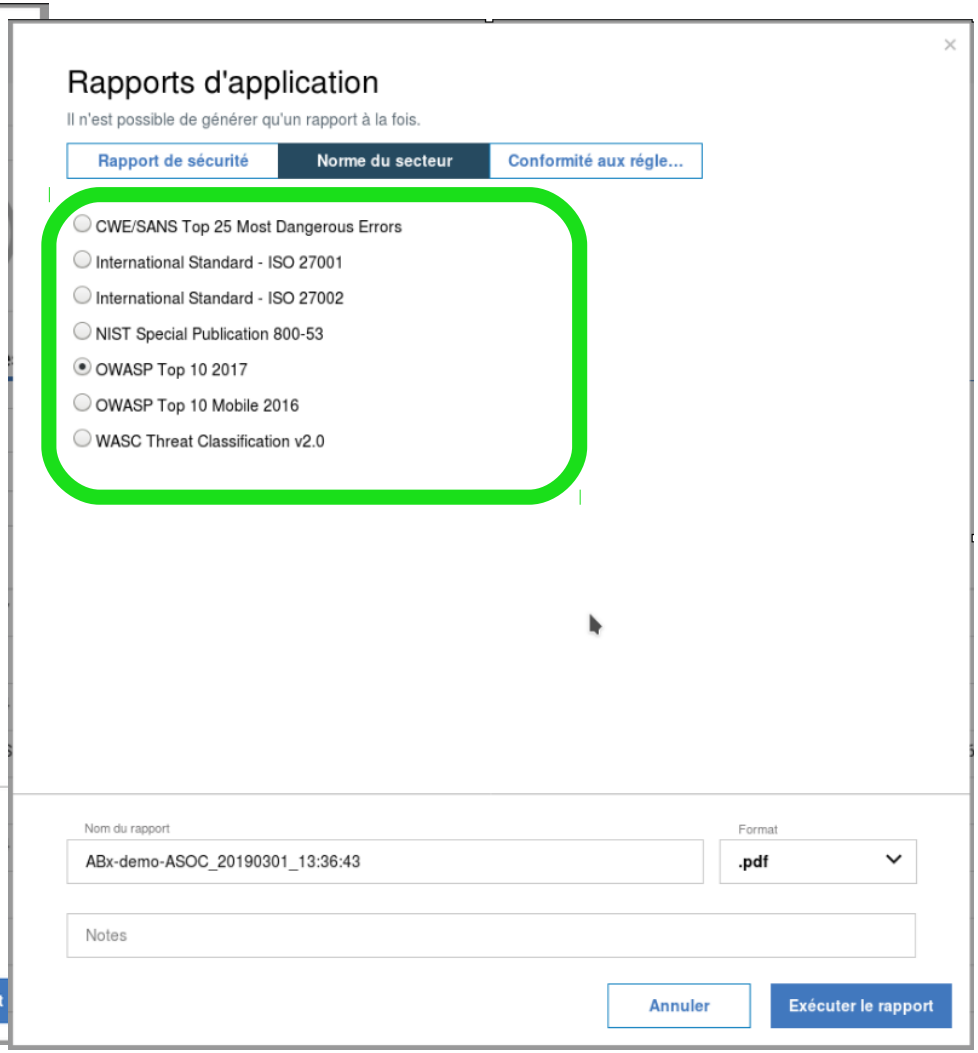
- Peu de configuration nécessaire à son exécution.
- Détermine les vecteurs d'attaque basés sur la politique choisie du test
- Génère un rapport incluant des conseils et des recommandations sur les corrections à réaliser.



Depuis ASoC – Choix de génération de rapports

- Les principaux standards sont disponibles





A votre disposition pour répondre à vos questions

ABlogiX

+33 2 85 29 43 43

arnaud.bourlier@ablogix.fr

www.ablogix.fr

@ABlogiX