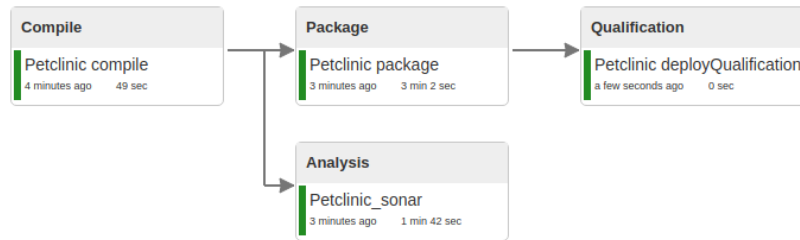


## Intégration dans Jenkins

But : Ajouter un item d'analyse Sonar en parallèle de l'item "**Petclinic\_package**"



## Installation du plugin Sonar Scanner

- Installer le plugin "**SonarQube Scanner**" (sans redémarrage)

### [SonarQube Scanner for Jenkins](#)



This plugin allows an easy integration of [SonarQube](#), the open source platform for Continuous Inspection of code quality.

[2.10](#)

- Configurer le plugin dans la section "**Administrer Jenkins > Configuration globale des outils > SonarQube Scanner**" (décocher *Install automatically*)
  - Name : SonarScanner
  - SONAR\_RUNNER\_HOME : /usr/share/sonar-scanner-4.2.0.1873

### SonarQube Scanner

Installations SonarQube Scanner

Ajouter SonarQube Scanner



SonarQube Scanner

Name

sonar scanner

SONAR\_RUNNER\_HOME

/usr/share/sonar-scanner-4.2.0.1873

☐ Install automatically

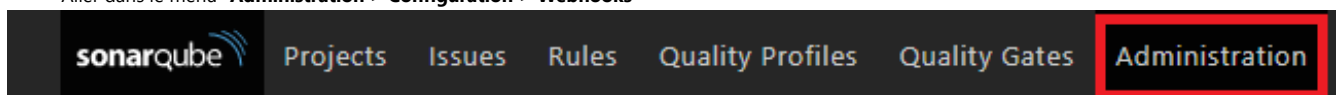


Supprimer SonarQube Scanner

Ajouter SonarQube Scanner

Liste des installations SonarQube Scanner sur ce système

- Dans **Sonarqube** :
  - Aller dans le menu "**Administration > Configuration > Webhooks**"



## Administration

**Configuration**

Security

Projects

System

Marketplace

General Settings

Encryption

Custom Metrics

**Webhooks**

this SonarQube instance.

- Créer un webhook avec les informations suivantes :
  - Name : Jenkins
  - URL : `http://[IP_SERVEUR_JENKINS]:8080/sonarqube-webhook`

- Secret : laisser vide

Administration

Configuration ▾ Security ▾ Projects ▾ System Marketplace

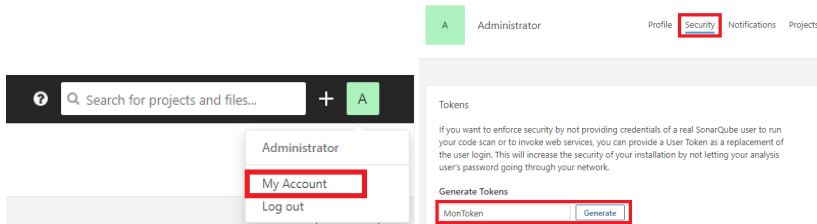
Webhooks

Webhooks are used to notify external services when a project analysis is done. An HTTP POST request including a JSON payload is sent to each of the provided URLs. Learn more in the [Webhooks documentation](#).

Create

Name	URL	Secret?	Last delivery
Jenkins	http://172.31.55.59:8080/sonarqube-webhook	No	December 12, 2019, 10:12 PM

- Aller dans le menu "My account > Security" et générer un token.



**!! ATTENTION !!** : copier le token généré pour ne pas le perdre, il sera impossible de le récupérer lorsque la page sera rechargée

## Generate Tokens

Enter Token Name

Generate



New token "MonToken" has been created. Make sure you copy it now, you won't be able to see it again!

Copy

ac320a8c23c0e560582a613e0044ee14bbf9adb5

- Dans **Jenkins** :
  - Configurer le server Sonarqube dans "**Administrer Jenkins > Configurer le système > SonarQube servers**" :
    - Nom : sonar
    - URL du serveur : http://[IP\_SERVEUR\_SONAR]:9000
    - Server authentication token : Ajouter un credential de type "**Secret text**" et coller le token créé précédement dans Sonarqube

### SonarQube servers

Environment variables

☐ Enable injection of SonarQube server configuration as build environment variables

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

Installations de SonarQube

Nom

sonar

URL du serveur

http://192.168.0.49:9000

Par défaut à http://localhost:9000

Server authentication token

sonarqube2

Ajouter

SonarQube authentication token. Mandatory when anonymous access is disabled.

Avancé...

Supprimer installation SonarQube

Ajouter une installation SonarQube

Liste des installations de SonarQube

## Création de l'item Petclinic\_sonar dans Jenkins

- Nom : Petclinic\_sonar
- Type : Projet free-style
- Paramètre string : WORKSPACE\_PARENT
- Actions :

1. Prends en paramètre le workspace du build parent (WORKSPACE\_PARENT)

☒ Ce build a des paramètres

Paramètre String

Nom

WORKSPACE\_PARENT

Valeur par défaut

Description

[Plain text] [Prévisualisation](#)

☐ Trim the string

Ajouter un paramètre ▾

2. Lance un build de type "**Lancer une analyse avec SonarQube Scanner**"

- Sélectionner le JDK "**java-1.8-openjdk**"
- Ajouter les propriétés suivantes :

```
sonar.java.binaries=target/classes
sonar.projectBaseDir=$WORKSPACE_PARENT
```

## Build

Lancer une analyse avec SonarQube Scanner

Tâche à lancer

JDK

java-1.8-openjdk ▾

Le JDK à utiliser pour cette analyse SonarQube

Chemin vers les propriétés du projet

Propriétés de l'analyse

sonar.java.binaries=target/classes  
sonar.projectBaseDir=\$WORKSPACE\_PARENT

Additional arguments

▾

Options de la JVM

▾

Ajouter une étape au build ▾

3. Modifier l'item "**Petclinic\_compile**" pour pouvoir lancer l'item "**Petclinic\_sonar**" en parallèle de l'item "**Petclinic\_package**"

Trigger parameterized build on other projects

Déclencheurs du build

Projects to build

Petclinic package

Trigger when build is

Stable

Trigger build without parameters

☐

Trigger build from child projects

☐

Predefined parameters

Parameters

WORKSPACE\_PARENT=\$WORKSPACE

Avancé...

Add Parameters

Projects to build

Petclinic\_sonar

Trigger when build is

Stable

Trigger build without parameters

☐

Trigger build from child projects

☐

Predefined parameters

Parameters

WORKSPACE\_PARENT=\$WORKSPACE

Avancé...

Add Parameters

Sauver

Apply

Ajouter un déclencheur...

- Lancer un build