

Installing and Configuring SonarQube Scanner for Jenkins

Table of Contents

- [Features](#)
- [Compatibility](#)
- [Prerequisites](#)
- [Installation](#)
- [Adding SonarQube Server](#)
 - [Enabling injection of SonarQube installations in Jenkins jobs as variables](#)
 - [Adding SonarQube Scanner](#)
 - [Adding SonarQube Scanner for MSBuild](#)

Features

You can trigger SonarQube analysis from Jenkins using either a:

- Build step to trigger the analysis with the SonarQube Scanner (preferred)
- Post-build action to trigger the analysis with Maven (SonarQube Scanner for Maven)

Compatibility

SonarQube Scanner for Jenkins	2.1	2.2.1	2.3
Jenkins	1.491+	1.580.1+	1.580.3+

Prerequisites

- SonarQube is already [installed](#)
- Jenkins is already [installed](#)
- Use at least the minimal version of Java supported by your SonarQube server

Installation

Install the SonarQube Scanner for Jenkins via the Jenkins Update Center.

Adding SonarQube Server

You can define as many SonarQube servers as you wish. Then for each Jenkins job, you will be able to choose which server to use for the SonarQube analysis.

To add a SonarQube server, just follow the three steps below:

1. Log into Jenkins as an administrator and go to [Manage Jenkins > Configure System](#):

? Unknown Attachment

2. Scroll down to the *SonarQube configuration* section and click on *"Add SonarQube"*.

? Unknown Attachment

3. To define more advanced settings for a SonarQube installation, click on the *"Advanced..."* button :

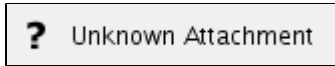
? Unknown Attachment

Enabling injection of SonarQube installations in Jenkins jobs as variables

Note that in on the top of the *SonarQube configuration* section, you can enable the ability of job administrators to inject in their builds the configuration of SonarQube instances as environment variables.

This allows them to use the SonarQube configurations in any type of jobs by using variables, such as Maven, Gradle, Ant, ...

Passwords are hidden from the build logs. By default, this feature is disabled.



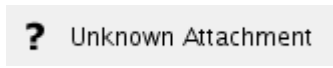
Adding SonarQube Scanner

This step is mandatory if you want to trigger any of your SonarQube analyses with the [SonarQube Scanner](#). Skip this step if you want to trigger all your analyses with [SonarQube Scanner for Maven](#).

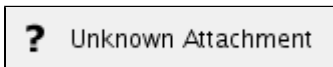
You can define as many SonarQube Scanner launchers as you wish. Then for each Jenkins job, you will be able to choose with which launcher to use to run the SonarQube analysis.

To add a SonarQube Scanner:

1. Log into Jenkins as an administrator and go to Manage Jenkins > Configure System:



2. Scroll down to the *SonarQube Runner* configuration section and click on *Add SonarQube Runner*. It is based on the typical Jenkins tool auto-installation. You can either choose to point to an already installed version of SonarQube Runner (uncheck 'Install automatically') or tell Jenkins to grab the installer from a remote location (check '*Install automatically*');



If you don't see a drop down list with all available SonarQube Runner versions but instead see an empty text field then this is because Jenkins still don't have downloaded required update center file (default period is 1 day). You may force this refresh by clicking 'Check Now' button in Manage Plugins >> Advanced tab.

Adding SonarQube Scanner for MSBuild

This step is mandatory if you want to trigger SonarQube analyses using the **SonarQube Scanner for MSBuild**.

You can define as **SonarQube Scanners for MSBuild** as you want. The tool can be automatically installed by **Jenkins**, or you can define the path where it is installed by unchecking "*Install automatically*".

For each Jenkins job, you will then be able to choose which launcher to use. If there is only one launcher defined, it will be implicitly selected.

In the [global configuration page](#), you can add installations of the **SonarQube Scanner for MSBuild**:

