SonarQube的安装和配置以及与Jenkins和gitlab-ci的集成

1. SonarQube安装前的准备
2. 安装JAVA（推荐OpenJDK1.8）

看文档装吧，懒得写

1. 安装Mysql（推荐5.6 & 5.7）

看文档装吧，懒得写

1. 安装PHP，PHPUNIT，XDEBUG

看文档装吧，懒得写

1. 用root身份执行如下命令

sysctl -w vm.max\_map\_count=262144

sysctl -w fs.file-max=65536

ulimit -n 65536

ulimit -u 2048

1. 永久修改系统参数

sudo vim /etc/sysctl.conf

或者 sudo vim /etc/sysctl.d/99-sonarqube.conf ，插入如下两行

sonarqube - nofile 65536

sonarqube - nproc 2048

sudo vim /etc/limits.conf

或者 sudo vim /etc/limits.d/99-sonarqube.conf ，插入如下两行

sonarqube - nofile 65536

sonarqube - nproc 2048

1. SonarQube安装并运行
   1. 下载并安装

wget <https://sonarsource.bintray.com/Distribution/sonarqube/sonarqube-7.2.1.zip>

unzip sonarqube-7.2.1.zip

sudo cp -r ./sonarqube-7.2.1 /usr/local/sonarqube

sudo vim /usr/local/sonarqube/conf/sonar.properties

插入或改写如下几行：

sonar.jdbc.username=root

sonar.jdbc.password=123456

sonar.jdbc.url=jdbc:mysql://127.0.0.1:3306/sonar?useUnicode=true&characterEncoding=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance&useSSL=false

请替换成自己的数据库IP、端口号以及用户名和密码

sonar.web.host=192.168.50.62

sonar.web.port=9000

请替换成自己的IP和端口号

* 1. 尝试运行

sudo adduser sonarUser

sudo passwd sonarUser

sudo chown -R sonarUser:sonarUser /usr/local/sonarqube

su sonarUser

/usr/local/sonarqube/bin/linux-x86-64/ sonar.sh console

用浏览器访问 http://192.168.50.62:9000

* 1. 安装服务

sudo ln -s /usr/local/sonarqube/bin/linux-x86-64/sonar.sh /usr/bin/sonar

sudo vim /etc/init.d/sonar

################################

#!/bin/sh

# rc file for SonarQube

# chkconfig: 345 96 10

# description: SonarQube system (www.sonarsource.org)

### BEGIN INIT INFO

# Provides: sonar

# Required-Start: $network

# Required-Stop: $network

# Default-Start: 3 4 5

# Default-Stop: 0 1 2 6

# Short-Description: SonarQube system(www.sonarsource.org)

# Description: SonarQube system (www.sonarsource.org)

### END INIT INFO

/usr/bin/sonar $\*

################################

sudo chmod 755 /etc/init.d/sonar

sudo chkconfig --add sonar

1. Sonar Scanner安装
   1. 下载并安装

wget https://sonarsource.bintray.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-3.2.0.1227-linux.zip

unzip sonar-scanner-cli-3.2.0.1227-linux.zip

vim ./sonar-scanner-3.2.0.1227-linux/conf/ sonar-scanner.properties

插入修改如下行，替换成自己的sonar服务器IP和端口号

sonar.host.url=http://192.168.50.62:9000

sudo cp -r ./sonar-scanner-3.2.0.1227-linux /usr/local/sonar-scanner

sudo ln -s /usr/local/bin/sonar-scanner /usr/local/sonar-scanner/bin/sonar-scanner

* 1. 项目实测

cd ~/nlp.xin.com

vim sonar-project.properties

###############################

#项目KEY

sonar.projectKey=nlp

#项目名称

sonar.projectName=NLP

#项目版本号

sonar.projectVersion=1.0

#需要审计的源代码目录

sonar.sources=./app

#项目根

sonar.projectBaseDir=./

#单元测试代码目录

sonar.tests=./tests

#项目编码字符

sonar.sourceEncoding=UTF-8

#项目编程语言

sonar.language=php

#单元测试覆盖率报告文件

sonar.php.coverage.reportPath=./logs/clover.xml

#单元测试报告文件

sonar.php.tests.reportPath=./logs/junit.xml

###############################

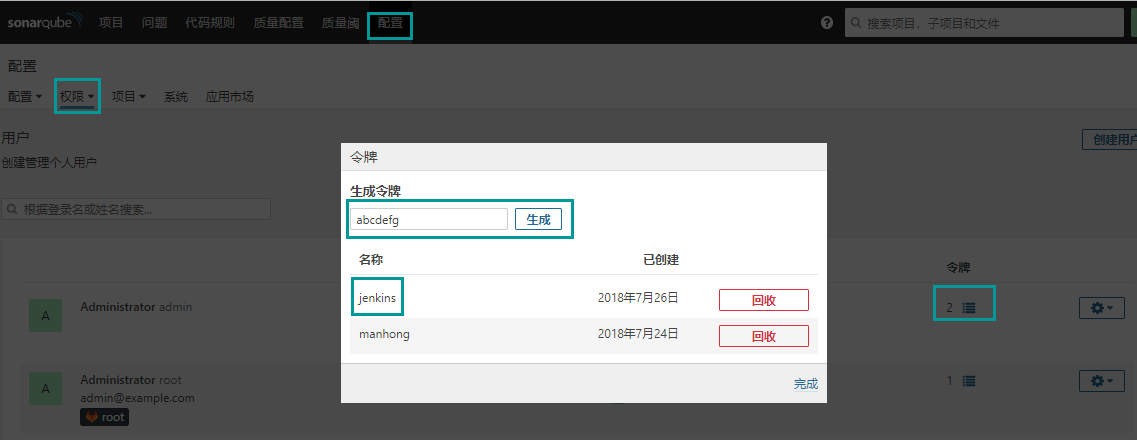
请替换成自己项目相关配置

sonar-scanner -X

或者如下（不需要配置文件，但请替换你自己的项目参数）：

sonar-scanner scan -X -Dsonar.host.url=http://192.168.50.62:9000 -Dsonar.sourceEncoding=UTF-8 -Dsonar.php.coverage.reportPath=./logs/clover.xml -Dsonar.sources=./app -Dsonar.language=php -Dsonar.projectVersion=1.0 -Dsonar.projectKey=nlp -Dsonar.php.tests.reportPath=./logs/junit.xml -Dsonar.projectBaseDir=./ -Dsonar.tests=./tests -Dsonar.projectName=NLP -Dsonar.login=cb67ba370c99c5577076aa9139764ffdddc54e0d

1. 通过Jenkins触发扫描
   1. 生成Sonar用户Token

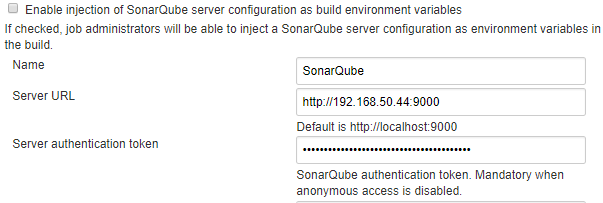


* 1. 安装并配置SonarQube Scanner for Jenkins插件

登录Jenkins => 系统管理 => 管理插件

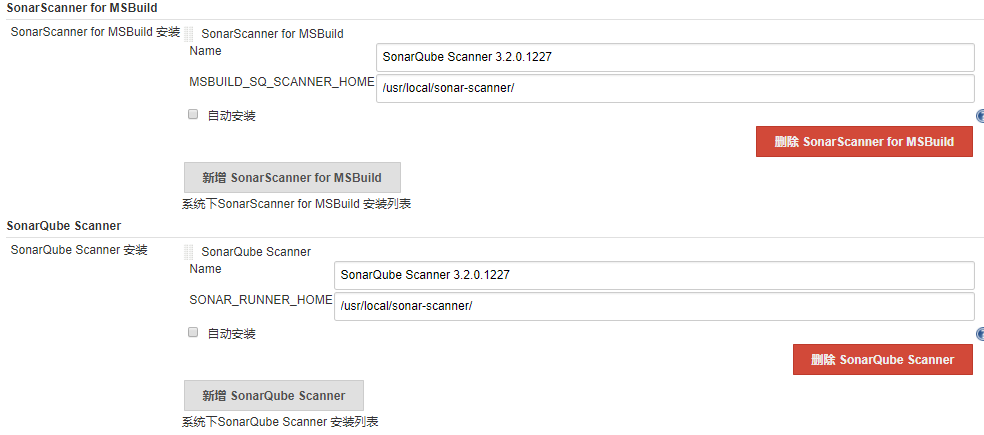


系统管理 => 系统设置 => SonarQube servers

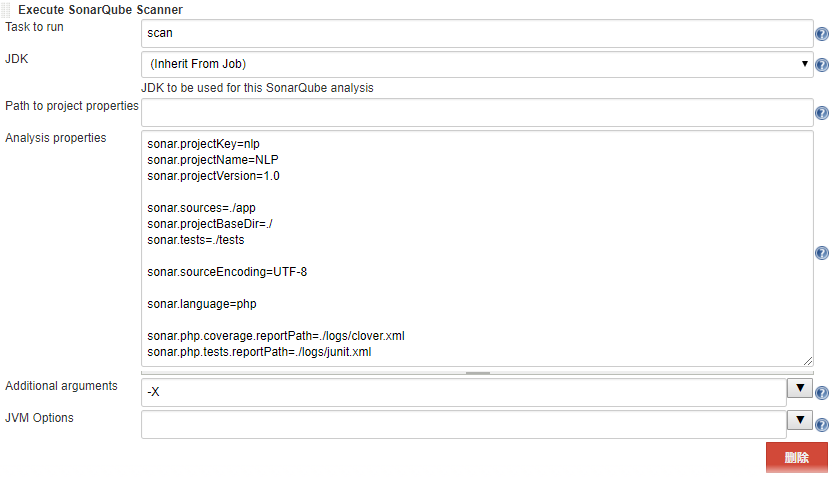


系在这里输入sonarQube服务器的ip、端口号，和刚才生成的Token

系统管理 => 全局工具配置 => SonarQube Scanner



NLP(具体项目) => 配置 => 构建 => Execute SonarQube Scanner



注意：Task to run配置项必须填scan，其他的根据自己项目情况而定。

1. 通过gitlab-ci触发扫描
2. 在安装了sonar-scanner的机器上安装gitlab-ci runner

sudo wget -O /usr/local/bin/gitlab-runner <https://gitlab-runner-downloads.s3.amazonaws.com/latest/binaries/gitlab-runner-linux-amd64>

sudo chmod +x /usr/local/bin/gitlab-runner

sudo useradd --comment 'GitLab Runner' --create-home gitlab-runner --shell /bin/bash

sudo gitlab-runner install --user=gitlab-runner --working-directory=/home/gitlab-runner

sudo gitlab-runner start

1. 注册gitlab-ci runner

首先找到项目用于注册runner的令牌

登录gitlab => projects(选择一个具体项目) => Settings => CI/CD => Runners settings



然后注册runner到gitlab

sudo gitlab-runner register

Please enter the gitlab-ci coordinator URL (e.g. https://gitlab.com )

http://gitlab.example.com/

Please enter the gitlab-ci token for this runner

z7CWgQeVx6TZDAkcxTJm

Please enter the gitlab-ci description for this runner

[hostame] sonar

Please enter the gitlab-ci tags for this runner (comma separated):

#标签，不用填，直接回车跳过

Please enter the executor: ssh, docker+machine, docker-ssh+machine, kubernetes, docker, parallels, virtualbox, docker-ssh, shell:

shell

1. 生成配置文件.gitlab-ci.yml，并放入项目版本库根目录中

before\_script:

#安装composer依赖，执行单元测，根据具体项目配置

- "/usr/local/php/bin/php /usr/local/bin/composer install"

- "/usr/local/php/bin/php /usr/local/bin/phpunit\_com"

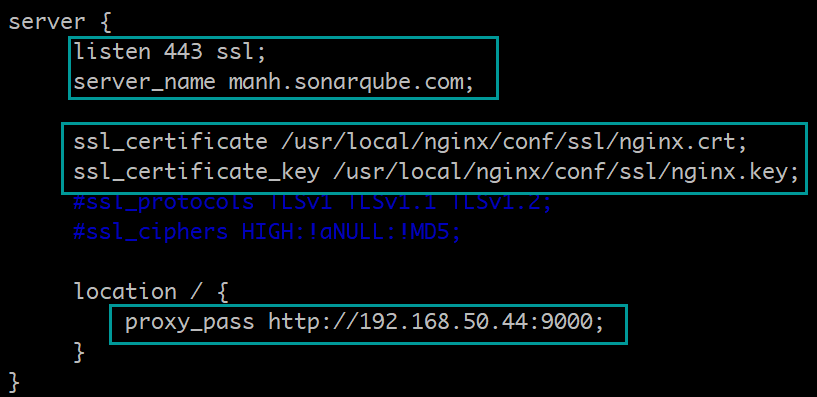
job:

script: "/usr/local/sonar-scanner/bin/sonar-scanner scan -X -Dsonar.host.url=http://192.168.50.44:9000 -Dsonar.sourceEncoding=UTF-8 -Dsonar.php.coverage.reportPath=./logs/clover.xml -Dsonar.sources=./app -Dsonar.language=php -Dsonar.projectVersion=1.0 -Dsonar.projectKey=nlp -Dsonar.php.tests.reportPath=./logs/junit.xml -Dsonar.projectBaseDir=./ -Dsonar.tests=./tests -Dsonar.projectName=NLP -Dsonar.login=cb67ba370c99c5577076aa9139764ffdddc54e0d"

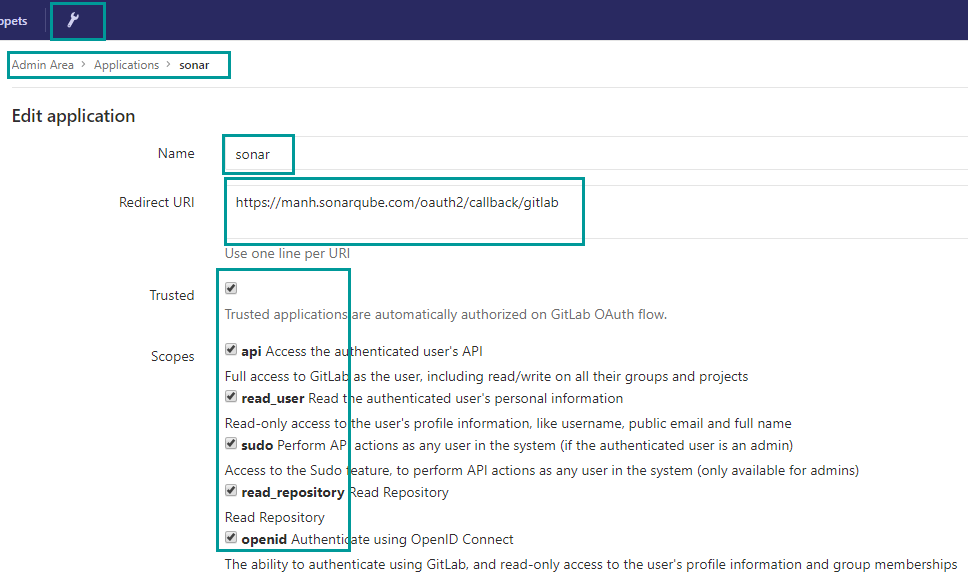
when: manual

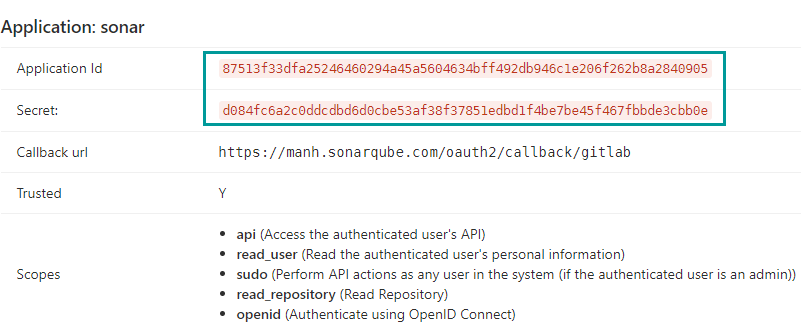
根据具体项目配置

1. 通过gitlab的oauth认证登录sonarQube
2. 如果需要通过gitlab进行oauth认证，那么sonarQube的回调地址必须是https的，怎么生成证书和秘钥自己查文档吧，简单把nginx配置粘上。



1. 生成gitlab的oauth 的 token

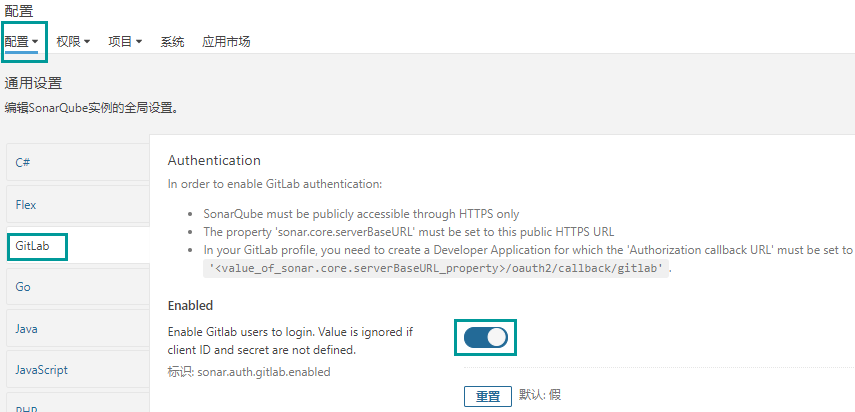




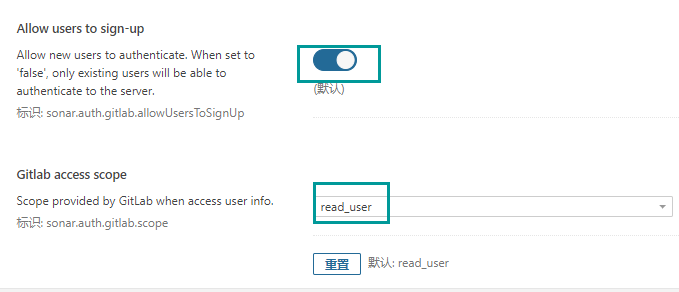
1. 安装并配置sonar-auth-gitlab-plugin 插件

wget <https://github.com/gabrie-allaigre/sonar-auth-gitlab-plugin/releases/download/1.3.2/sonar-auth-gitlab-plugin-1.3.2.jar>

cp sonar-auth-gitlab-plugin-1.3.2.jar /usr/local/sonarqube/extensions/plugins/ sonar-auth-gitlab-plugin-1.3.2.jar









这里要注意，对于旧版本的gitlab，比如8.X版本，gitlab access scope 选项要填 none，而不是 read\_user.