

gitlab-jenkins-sonar

一 环境

使用trustie-mysql和gitlab-jenkins-sonar容器【[zd维护](#)】

镜像导入：

```
cat gitlab-jenkins-sonar.tar | docker import - gitlab-jenkins-sonar
cat trustie-mysql.tar | docker import - trustie-mysql
```

trustie-mysql容器启动：

```
docker run -itd --name trustie-mysql --privileged -p 8888:8080 -p 1122:22 --restart=always trustie-mysql /home/pdl/start.sh
```

gitlab-jenkins-sonar容器启动：

```
docker run -itd --name gitlab-jenkins-sonar --privileged -p 8889:8080 -p 8890:8082 -p 8891:9000 -p 1123:22 --link trustie-mysql:gitlabmysql --restart=always gitlab-jenkins-sonar /home/git/start.sh
```

进入容器启动服务【[需要自重启脚本](#)】：

```
docker exec -it gitlab-jenkins-sonar bash
#sonarqube启动
root@9dac39681d8c:/# /usr/local/sonarqube-5.2/bin/linux-x86-64/sonar.sh restart
#jenkins 启动
service jenkins start
```

注意：物理机内存不要太小，否则会导致sonar或jenkins服务停止
内存》3G

端口对应关系：

:8889是gitlab
:8890是jenkins
:8891是sonar

二 配置

jenkins配置:

1、先装插件，在8890/pluginManager/

Build Authorization Token Root Plugin

Git plugin
GitLab Plugin
Gitlab Hook Plugin
Gitlab Authentication plugin

SonarQube Plugin

2、系统配置

[Jenkins](#)=>Manage Jenkins=>Configure System (:8890/configure)

下面的需要配置Sonar相关的东西，包括server和scanner

sonarQube servers	<input checked="" type="checkbox"/> Enable injection of SonarQube server configuration as build environment variables <small>If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.</small>
Environment variables	
SonarQube installations	
Name	SonarQube 5.2
Server URL	http://192.168.8.158:8891/
Server version	5.2 <small>Default is http://localhost:9000</small>
Server authentication token	<input type="text"/>
SonarQube account login	admin <small>SonarQube authentication token. Mandatory when anonymous access is disabled.</small>
SonarQube account password	***** <small>SonarQube account used to perform analysis. Mandatory when anonymous access is disabled. No longer used since SonarQube 5.3.</small>
	<small>SonarQube account used to perform analysis. Mandatory when anonymous access is disabled. No longer used since SonarQube 5.3.</small>

SonarQube Scanner

SonarQube Scanner installations

SonarQube Scanner

Name: SonarRunner

SONAR_RUNNER_HOME: /usr/local/sonar-runner-2.4

Install automatically

Delete SonarQube Scanner

Add SonarQube Scanner

List of SonarQube Scanner installations on this system

还要配置下Gitlab :

Gitlab

GitLab connections

Connection name: trustie-gitlab

Gitlab host URL: http://192.168.8.159:8889

API Token: GitLab API Token **Add**

API Token for Gitlab access required

Ignore SSL Certificate Errors

Test Connection

Delete

Add

3、手动集成

(1) 先在jenkins新建job，使用freestyle project

Jenkins

Jenkins > All >

New Item

Item name: myjob

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can

Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

External Job
This type of job allows you to record the execution of a process run outside Jenkins, even on a remote machine. This is des See the [documentation for more details](#).

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform

Copy existing Item
Copy from

OK

project name是必须的，gitlab connection选择system config配置的那个

Project name: myjob

Description: 请问您今天要来点兔子吗?

Discard Old Builds

GitLab connection: trustie-gitlab

This build is parameterized

Disable Build (No new builds will be executed until the project is re-enabled.)

Execute concurrent builds if necessary

source code management选择git，并填写要集成测试的git仓库地址 (Gitlab Repo的地址)

Source Code Management

- None
- CVS
- CVS Projectset
- Git

Repository URL

Credentials

Branches to build

Branch Specifier (blank for 'any')

Repository browser

Additional Behaviours

- Subversion

public仓库不需要credentials, 如果是private私有仓库, 需要add验证

触发器使用 Gitlab那个, 然后记下 Gitlab CI Service URL

Build Triggers

Build after other projects are built

Build periodically

Build when a change is pushed to GitLab 这个是webhook的通知地址

Build on Merge Request Events

Build on Push Events

Rebuild open Merge Requests

Enable [ci-skip]

Set build description to build cause (eg. Merge request or Git Push)

Add note with build status on merge requests

Vote added to note with build status on merge requests

Accept merge request on success

Allow all branches to trigger this job

Filter branches by name

Filter branches by regex

Poll SCM

构建时候选SonarQube Scanner

Add build step

Execute SonarQube Scanner

Execute Windows batch command

Execute shell

Execute shell script on remote host using ssh

Invoke Ant

Invoke top-level Maven targets

SonarQube Scanner for MSBuild - Begin Analysis

SonarQube Scanner for MSBuild - End Analysis

之后出现scanner的build列, 配置build过程, 填写Analysis properties (必须), 其他的为可选参数 (比如 运行JVM参数)

Build

Execute SonarQube Scanner

Task to run

JDK (Inherit From Job)
JDK to be used for this sonar analysis

Path to project properties

Analysis properties

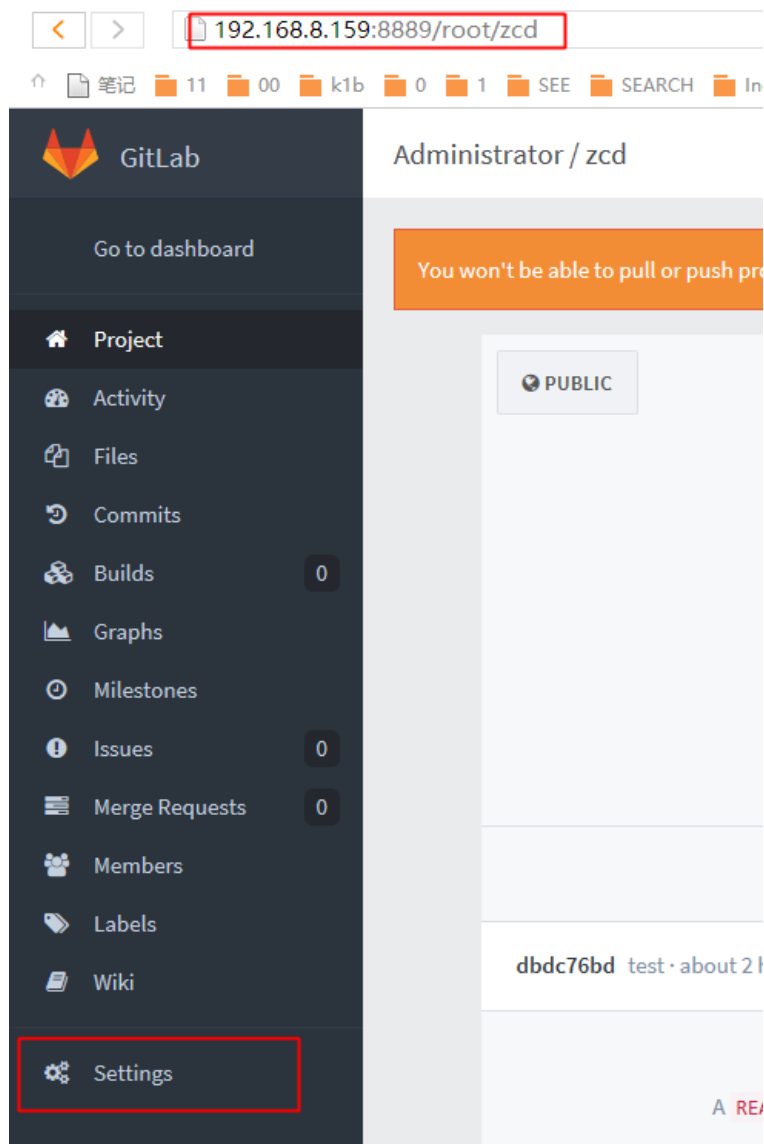
```
sonar.projectKey=my:project
sonar.projectName=My project
sonar.projectVersion=1.0
sonar.sources=src/main/java
sonar.language=java
```

Additional arguments

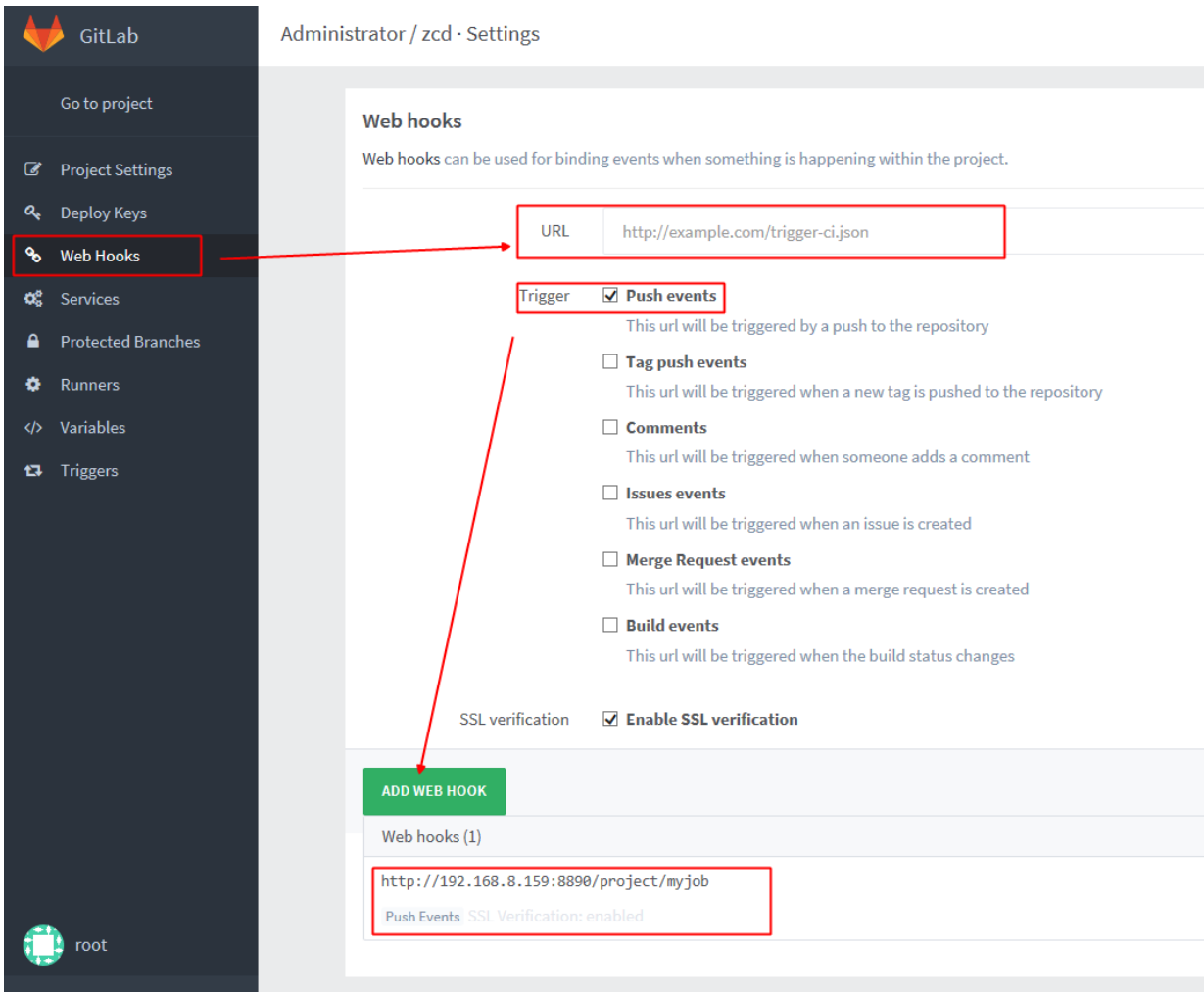
JVM Options

补充：加上sonar.sourceEncoding=utf-8,防止中文乱码

(2) 配置gitlab的webhook
 进入Gitlab，进入要基础测试的项目（上面git url）



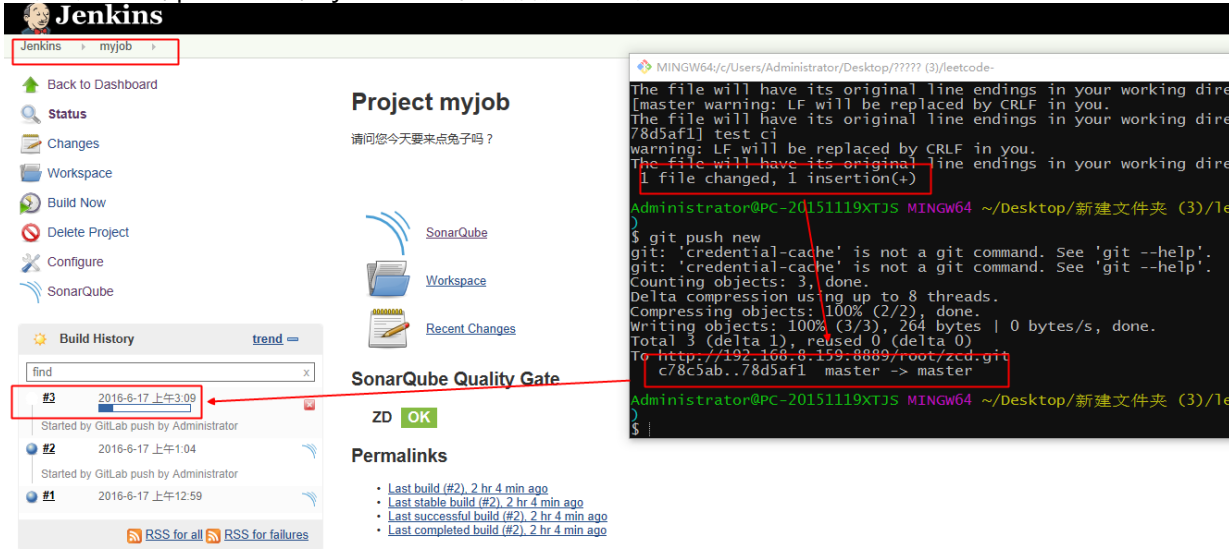
在webhook里，添加一个push events的hook，把上面再jenkins job里，触发器给的地址加入进去



到此，gitlab-jenkins-sonar的持续集成测试搞好了。

(3) 测试

在本地修改提交，push成功后，在job的status中自动启动build过程：



点击build的编号#3,进到详情页：

在console output中可以看到具体的执行过程输出

Jenkins #3

Back to Project, Status, Changes, Console Output, View as plain text, Edit Build Information, Delete Build, Polling Log, Git Build Data, No Tags, Previous Build

Console Output

```

Started by Gitlab push by Administrator
Building in workspace /var/lib/jenkins/jobs/myjob/workspace
Unable to auto-install JDK until the license is accepted.
Unable to auto-install JDK until the license is accepted.
> git rev-parse --is-inside-work-tree # timeout=10
> git config remote.origin.url http://192.168.8.159:8889/root/zcd.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url http://192.168.8.159:8889/root/zcd.git
Retrieving upstream changes from http://192.168.8.159:8889/root/zcd.git
> git --version # timeout=10
> git -c core.askpass=true fetch --tags --progress http://192.168.8.159:8889/root/zcd.git +refs/heads/*:refs/remotes/origin/*
skipping resolution of commit 78d5af116149561669b037e45403cfe7d808cc6, since it originates from another repository
> git rev-parse refs/remotes/origin/master#{commit} # timeout=10
> git rev-parse refs/remotes/origin/master#{commit} # timeout=10
Unable to auto-install JDK until the license is accepted.
Checking out Revision 78d5af116149561669b037e45403cfe7d808cc6 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f 78d5af116149561669b037e45403cfe7d808cc6
> git rev-list dbdc76bd54626732b9b6434a3e5e926c591f293 # timeout=10
Unable to auto-install JDK until the license is accepted.
[workspace] # /usr/local/sonar-runner-2.4/bin/sonar-runner -e -Dsonar.host.url=http://192.168.8.159:8891 -Dsonar.projectName=ZD -Dsonar.projectVersion=1.22 -Dsonar.sourceEncoding=utf-8 -Dsonar.projectKey=root:zcd -Dsonar.sources=src
-Dsonar.projectBaseDir=/var/lib/jenkins/jobs/myjob/workspace
SonarQube Runner 2.4
INFO: Runner Corporation (64-bit)
INFO: Linux 3.13.0-87-generic amd64
INFO: Error stacktraces are turned on.
INFO: Runner configuration file: /usr/local/sonar-runner-2.4/conf/sonar-runner.properties
INFO: Project configuration file: NONE
INFO: Default locale: "en_US", source code encoding: "utf-8"
INFO: Work directory: /var/lib/jenkins/jobs/myjob/workspace/.sonar
INFO: SonarQube Server 5.2
03:09:44.490 INFO - Load global repositories
03:09:44.826 INFO - Load global repositories (done) | time=342ms
03:09:44.848 WARN - Property 'sonar.jdbc.url' is not supported any more. There is no more DB connection to the SQ database. It will be ignored.
03:09:44.852 INFO - User cache: /var/lib/jenkins/.sonar/cache
03:09:45.656 INFO - Load plugins index
03:09:45.665 INFO - Load plugins index (done) | time=9ms
03:09:46.338 INFO - Process project properties
03:09:46.920 INFO - Load project repositories
03:09:49.208 INFO - Load project repositories (done) | time=288ms
03:09:47.249 INFO - Apply project exclusions
03:09:47.322 INFO - Load quality profiles
03:09:47.460 INFO - Load quality profiles (done) | time=138ms
03:09:47.463 INFO - Load active rules
03:09:59.666 INFO - Analysis reports sent to server in 244ms
03:09:59.668 INFO - ANALYSIS SUCCESSFUL, you can browse http://192.168.8.159:8891/dashboard/index/root:zcd
03:09:59.668 INFO - Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report.
03:09:59.668 INFO - More about the report processing at http://192.168.8.159:8891/api/ce/task?id=AVVcVi282EX5PrdM4wab
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
Total time: 16.272s
Final Memory: 8M/242M
INFO: -----
Finished: SUCCESS

```

执行sonar-scanner扫描完成后

192.168.8.159:8891

sonarqube Dashboards Issues Measures Rules Quality Profiles Quality Gates Administration More

Home

Welcome to SonarQube Dashboard

Since you are able to read this, it means that you have successfully started your SonarQube server. Well done!

If you have not removed this text, it also means that you have not yet played much with SonarQube. So here are a few pointers for your next step:

- Do you now want to run analysis on a project?
- Maybe start customizing dashboards?
- Or simply browse the complete documentation?
- If you have a question or an issue, please visit the Get Support page.

MY FAVOURITES

QG NAME	LAST ANALYSIS
No data	

PROJECTS

QG NAME	VERSION	LOC	TECHNICAL DEBT
My project	1.0	54	1h
My project122	1.23	58,647	0
My project211	1.22	58,647	0
My project985	1.11	2,159	8d 5h
today2	1.4	3,851	4d 2h
ZD	1.22	2,159	8d 5h

6 results

sonarqube Dashboards Issues Measures Rules Quality Profiles Quality Gates Administration More

Administrator

June 17, 2016 11:09 AM Version 1.1

Main Dashboard Components Issues Administration More

Main Dashboard

Time changes... Configure widgets

Lines Of Code 2,159	Files 46	Functions 148	SOALE Rating A	Technical Debt Ratio 6.4%
Java Directories: 3 Lines: 2,861	Classes: 59 Statements: 1,237 Accessors: 8		Debt: 8d 5h	Issues: 503

Duplications
14.9%

Lines: 426, Blocks: 14, Files: 5

Complexity
515

/Function: 3.5, /Class: 8.7, /File: 11.2

Events: All

17 Jun 2016 Version: 1.22

The project has passed the quality gate.