jenkins 持续部署 CD

前言

在测试行业打滚了四年了,还没有接触过 cicd 的东西,刚好有时间就学习起来了,因此有了这篇学习笔记。简单地介绍本文的内容,这边文章主要记录了本人从环境准备,到 sonarqube 简单集成,最后达到简单的持续部署的实践过程。

一、安装

使用 vmware 安装 centos7 虚拟机,安装过程就省略了。

配置桥接模式:

- 1. 在虚拟机设置-网络适配器,选择桥接模式,并勾选 "复制物理网络连接状态"
- 2. 检查 vmware 中是否有桥接网络,在 vmware 的菜单-编辑->虚拟网络编辑器,查看桥接类型的网络是否存在。若不存在,需要点击桥接模式,选择本地物理网卡,点击添加网络即可

安装 jenkins:

- 1. 打开虚拟机的终端, 切换到 root, 运行以下命令:
- sudo wget -0 /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat/jenkins.repo
- rpm import https://pkg.jenkins.io/redhat/jenkins.io.key
- yum install jenkins
- 2. 由于使用这种方式安装,jenkins 会自动注册成 linux 服务,可以通过 systemctl start jenkins 或 systemctl stop jenkins 启动或停止服务
- 3. 确认 jenkins 服务启动后,打开浏览器 localhost:8080 就可以见到老头了。如果在别的机器上无法访问,十有八九是防火墙的问题。使用 systemctl stop firewalld 关闭防火墙即可
- 4. 初始登录密码可以在/var/lib/jenkins/secrets/initialAdminPassword 中找到。如果不知道这个文件放哪里,可以运行以下命令找到 JENKINS_HOME, 然后再找到首次密码文件

[root@localhost secrets]# ps -ef | grep jenkins

jenkins 11083 1 1 16:54 ? 00:00:26 /etc/alternatives/java -Dcom. sun. akuma. Daemon=daemonized -Djava. awt. headless=true -

DJENKINS HOME=/var/lib/jenkins -jar /usr/lib/jenkins/jenkins.war --logfile=/var/log/jenkins/jenkins.log --

webroot=/var/cache/jenkins/war --daemon --httpPort=8080 --debug=5 --handlerCountMax=100 --handlerCountMaxIdle=20

5. 选择了安装建议的插件,完成后就可以看到首页了。

创建项目

工具准备

点击系统管理->Global Tool Configuration,配置 JDK Git 和 Maven。

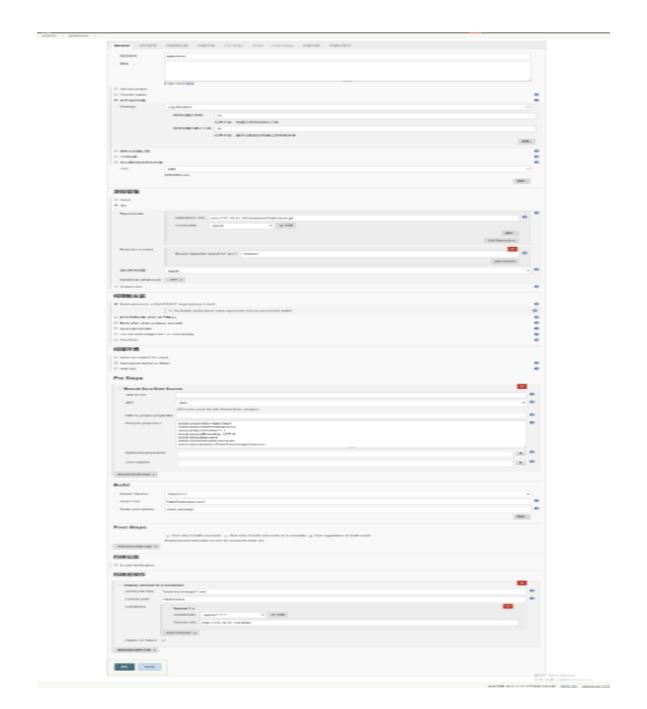
I .	-	
JDK		
JDK 安装	■ JDK	
	别名 jdk9	
	☑ 自动安装	0
	从 java.sun.com安装 版本 Java SE Development Kit 9.0.1	
	■ 我同意 Java SE Development Kit 的许可协议	
		删除安装
	新增安装 ▼	
		删除JDK
	JDK 別名 idks	
	Juko	
	JAVA_HOME /usr/java/jdk1.8.0_111	
	■ 自动安装	•
		删除 JDK
	新增 JDK	
	系统下JDK 安装列表	
Git		
Git installations		
	☐ Git	
	Name Default	
	Path to Git executable /usr/hin/nit	
	Path to Git executable /usr/bin/git	•
	□ 自动 安装	②
		Delete Git
		Detete Git

Maven		
Maven 安装	Maven	
	Name maven3.5	
	● 自动安装	•
	Install from Apache	
	版本 3.5.2 ▼	删除安装
	新增安装 ▼	
		删除 Maven
	Maven	
	Name maven2.1	
	■ 自动安装	•
	Install from Apache	
	版本 2.1.0 ▼	
		删除安装
	新增安装 ▼	
		删除 Maven
	新增 Maven 查看原图	
	系统下Maven 安装列表	

MAVEN 的配置

备注: jdk8 和 git 是事先在服务器中安装完成的。 **新建**

点击新建按钮,填写项目名称并选择 maven 项目,进入配置界面,并配置为下图所示如下所示:



taskcheck 的配置界面

说明:

1. 在配置 ssh 方式 git 的服务器时,需要注意权限问题。我这里是先用组 gituser 下的 xiemb 用户创建一个文件夹,然后运行以下命令,设置该目录 以及该目录下的文件对于组 gituser 下的用户都是 rwx。

\$mkdir TaskCheck.git

\$chmod g=rwx TaskCheck.git

\$ su root

setfacl -m d:g:gituser:rwx /newdata/TaskCheck.git

2. 配置 git 用户时,需要事前安装 SSH Credentials Plugin 插件,并且获取对应用户的公钥。

```
[xiemb@gitServer .ssh]$ pwd
/home/xiemb/.ssh
[xiemb@gitServer .ssh]$ ls
authorized_keys known_hosts
[xiemb@gitServer .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaClycZEAAAADAQABAAABAQDIjtsoMwZIClYNEdFoEDTOyxUSdxzYmsqNKwyY9T/h
vbrASbIbnA+qAmlRDZ5WsL6A9eMj9tQO6dLeWrm9yJkfKpbe7sZ464R8RYKzbnJuUvq76KxQyBoaZ8bx
M41zC3iQLCGQx4ZDDL8W+s7pkZ42YFPNkyr96Imw8C0k/YxNeHPHDcubfUNcX1+wngBZROUujzB/yzDF
6DZiLVoXfBJuknsjF9Y4yYouznrxH/CmZnR3V1SD6qwwFGs2J25MKsSijzQkkhF3LclCae7UqgK1NAHz
dNUKuRDb6qprq27wsB3yWcuQXy5h354Ts76kO9HyoElgU5KLK3Cy10FCV1K/ gitserver@gitServer
[xiemb@gitServer .ssh]$
```

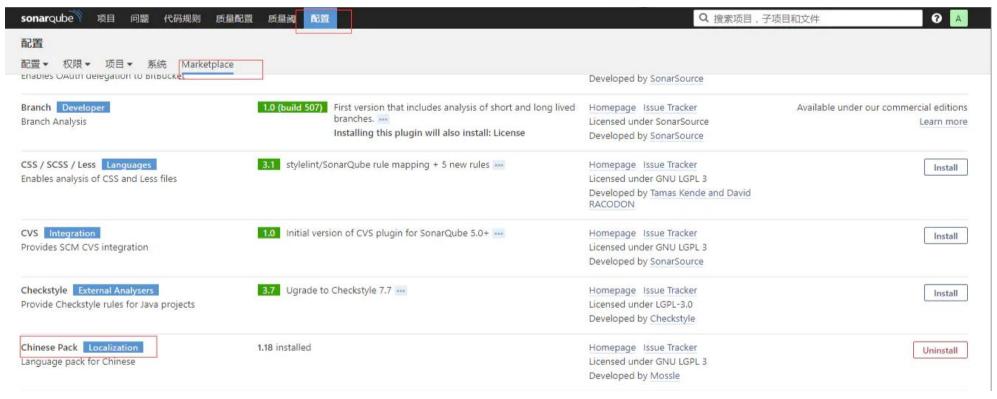
用户 xiemb 的 ssh 公钥

- 3. 配置 Sonar Quebe Scanner,需要下载相关组件,并且需要配置好 sonar 服务器。
- 3.1jenkins 插件下载: Sonargraph Integration Jenkins Plugin、Sonargraph Plugin 和 SonarQube Scanner for Jenkins。
- 3.2 在 <u>sonar 官网</u>下载安装包,解压后,配置\$SONAR_HOME/conf/sonar.properties 文件,我只是配置了数据库,启动服务\$SONAR HOME/bin/linux-x86-64/sonar.sh start。

```
# User credentials.
# Permissions to create tables, indices and triggers must be granted to JDBC use r.
# The schema must be created first.
sonar.jdbc.username=sonardba
sonar.jdbc.password=sonardba
#---- Embedded Database (default)
# H2 embedded Database server listening port, defaults to 9092
#sonar.embeddedDatabase.port=9092
#---- MySQL 5.6 or greater
# Only InnoDB storage engine is supported (not myISAM).
# Only the bundled driver is supported. It can not be changed.
sonar.jdbc.url=jdbc:mysql://172.16.31.129:3306/sonardb?useUnicode=true&character
Encoding=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance&useSSL=fal
se
```

sonar. properties 部分配置

3.3 访问 http://ip:9000 就会看到 sonar 的界面了,点击登陆,输入 admin/admin 后,进入配置,marketplace,安装 Chinese pack。



安装中文包

3.4 到此, sonar 的服务端算是搞掂了, 然后回到 taskcheck 作业的配置界面, 配置 execute sonarqube scanner。PS: 如果 projectVersion 从 pom. xml 获取变成参数会更佳。



sonar 配置

4.. 项目构建后部署到 tomcat 中,需要配置具有部署权限的 tomcat user。

```
<role rolename="manager-gui" />
<role rolename="manager-script" />
<user username="admin" password="Aa123456" roles="manager-gui, manager-script" />
```

tomcat-user.xml 部分配置

最大的坑

由于 TaskCheck 的架构是 Spring+Springmvc+mybatis+mysql, 而 mybatis 的映射文件放在源码某个包中,导致打包时没有把映射文件*.xml 放入相应的文件。

```
2017-11-10 17:46:17.666 [http-bio-8080-exec-2] WARN - Exception encountered during context initialization - cancelling refresh attempt: org.springframework.beans.factory.UnsatisfiedDependencyE
ckService': Unsatisfied dependency expressed through field 'enterpriseDao'; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'IEnterpriseDao';
ory.BeanCreationException: Error creating bean with name 'sqlSessionFactory' defined in class path resource [spring-core.xml]: Initialization of bean failed; nested exception is org.springframew
convert property value of type 'java.lang.String' to required type 'org.springframework.core.io.Resource[]' for property 'mapperLocations'; nested exception is java.lang.IllegalArgumentException rn [classpath:com/www.mapping/*.xml]: class path resource [com/www.mapping/] cannot be resolved to URL because it does not exist
2017-11-10 17:46:17.667 [http-bio-8080-exec-2] ERROR - Context initialization failed
org.springframework.beans.factory.UnsatisfiedDependencyException: Error creating bean with name 'checkService': Unsatisfied dependency expressed through field 'enterpriseDao'; nested exception i
ationException: Error creating bean with name 'IEnterpriseDao' defined in file [/home/xiemb/apache-tomcat-7.0.72/webapps/TaskCheck/WEB-INF/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes/com/webaps/classes
tion of bean failed; nested exception is org.springframework.beans.TypeMismatchException: Failed to convert property value of type 'java.lang.String' to required type 'org.springframework.core.i
 ; nested exception is java.lang.IllegalArgumentException: Could not resolve resource location pattern [classpath:com/xuanwu/mapping/*.xml]: class path resource [com/xuanwu/mapping/*.xml]: class path resource [com/xuanwu/mapping/*.xml]:
                                                                                                                                                                                                                                                                  /mapping/] cannot be r
           at org.springframework.beans.factory.annotation.AutowiredAnnotationBeanPostProcessor$AutowiredFieldElement.inject(AutowiredAnnotationBeanPostProcessor.java:588)
           at org.springframework.beans.factory.annotation.InjectionMetadata.inject(InjectionMetadata.java:88)
           at org.springframework.beans.factory.annotation.AutowiredAnnotationBeanPostProcessor.postProcessPropertyValues(AutowiredAnnotationBeanPostProcessor.java:366)
           at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.populateBean(AbstractAutowireCapableBeanFactory.java:1264)
           at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.doCreateBean(AbstractAutowireCapableBeanFactory.java:553)
           at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.createBean(AbstractAutowireCapableBeanFactory.java:483)
           at org.springframework.beans.factory.support.AbstractBeanFactory$1.getObject(AbstractBeanFactory.java:306)
           at \ org.springframework.beans.factory.support.DefaultSingletonBeanRegistry.getSingleton(DefaultSingletonBeanRegistry.java:230) \\
           at org.springframework.beans.factory.support.AbstractBeanFactory.doGetBean(AbstractBeanFactory.java:302)
           at org.springframework.beans.factory.support.AbstractBeanFactory.getBean(AbstractBeanFactory.java:197)
           at org.springframework.beans.factory.support.DefaultListableBeanFactory.preInstantiateSingletons(DefaultListableBeanFactory.java:761)
           at org.springframework.context.support.AbstractApplicationContext.finishBeanFactoryInitialization(AbstractApplicationContext.java:867)
           at org.springframework.context.support.AbstractApplicationContext.refresh(AbstractApplicationContext.java:543)
            at org.springframework.web.contextLoader.configureAndRefreshWebApplicationContext(ContextLoader.java:443)
```

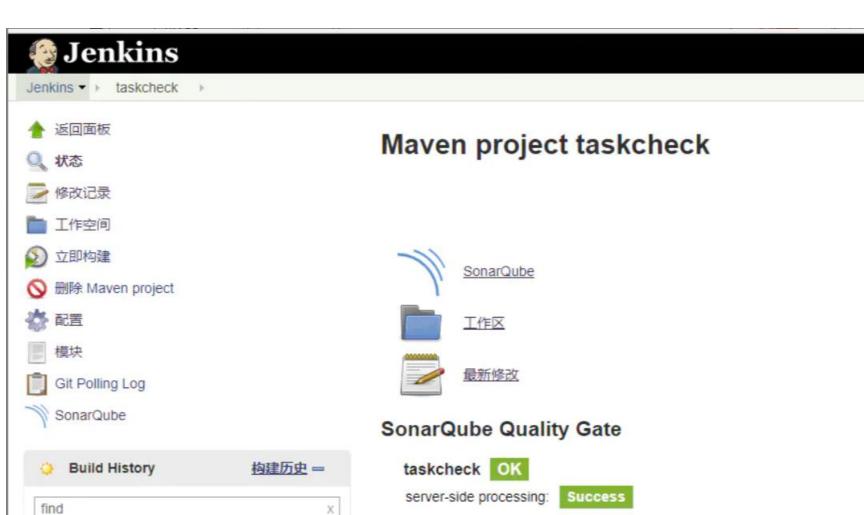
tomcat 报错

网上是有不少方法,但这个项目是配置在某个包下,而不是编译路径,导致我花费了不少的时间去尝试,最后的解决方法也很简单。

```
<build>
  <sourceDirectory>src</sourceDirectory>
  <resources>
   <resource>
     <directory>configure</directory>
     <excludes>
       <exclude>**/*.java</exclude>
     </excludes>
   </resource>
   <resource>
     <directory>src/com/mapping</directory>
     <filtering>true</filtering>
      <targetPath>${project.build.directory}/classes/com/mapping</targetPath>
     <includes>
         <include>*.xml</include>
     </includes>
    </resource>
  </resources>
```

在项目的 pom. xml 配置 resource

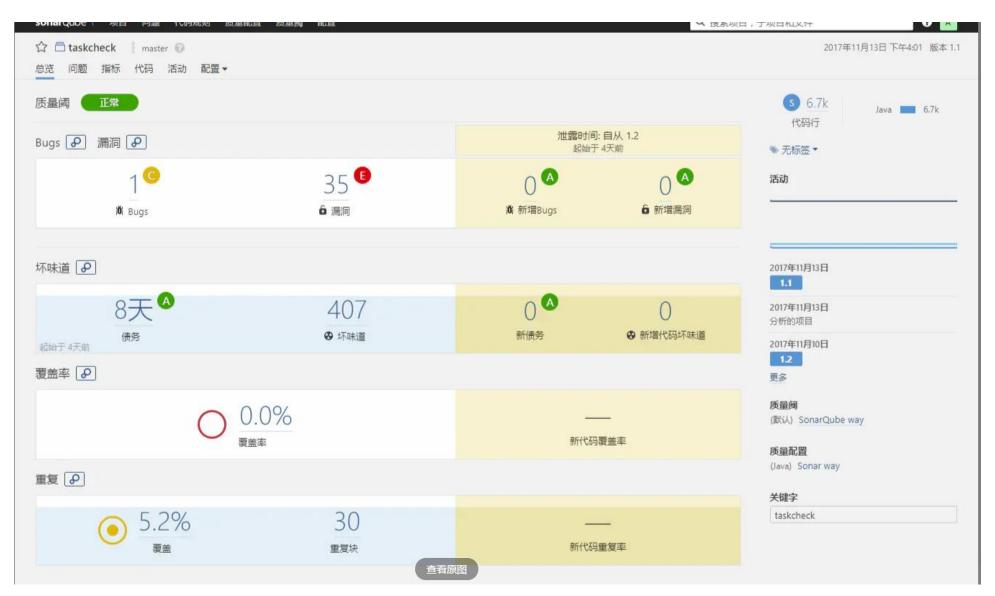
展示



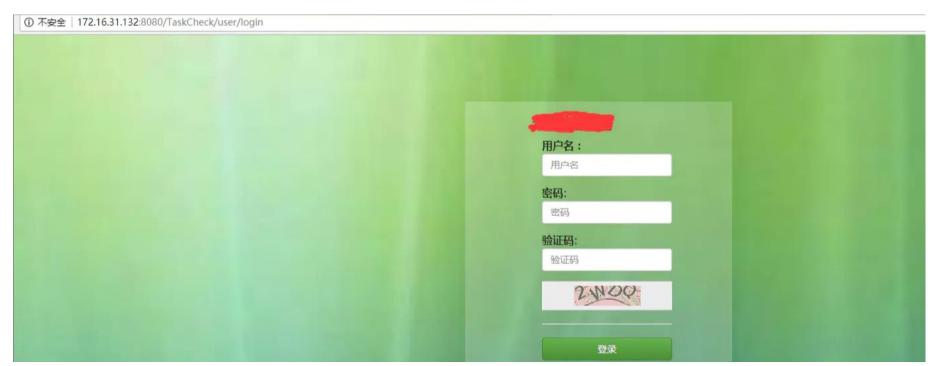


相关连接

- Last build(#34),18 小时之前
- · Last stable build(#34).18 小时之前
- Last successful build(#34),18 小时之前
- · Last failed build(#29),3 days 15 小时之前
- Last unsuccessful build(#29),3 days 15 小时之前
- · Last completed build(#34),18 小时之前 查看原图



SonarQube 扫描结果



访问应用程序

写在最后

由于本人也是刚接触 jenkins, 如有错漏,请指出,我会尽快修改过来。如有疑问,欢迎交流。