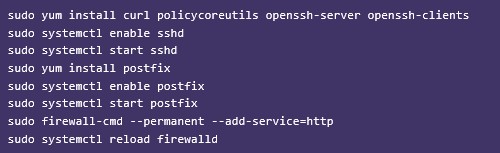
**Gitlab + Jenkins + Sonarqube + ZenTao 环境搭建及操作手册**

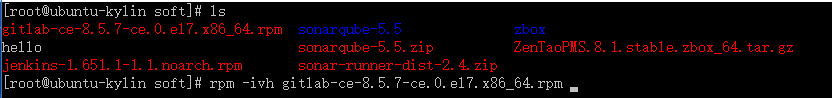
1. **GitLab的安装**
2. **安装和配置必要的依赖项**



1. **下载安装包**

**从url =** [**https://packages.gitlab.com/gitlab/gitlab-ce/packages/el/6/gitlab-ce-8.7.5-ce.0.el6.x86\_64.rpm**](https://packages.gitlab.com/gitlab/gitlab-ce/packages/el/6/gitlab-ce-8.7.5-ce.0.el6.x86_64.rpm)**下载最新的安装包，上传到服务器相应的文件夹中；**

1. **安装gitlab**

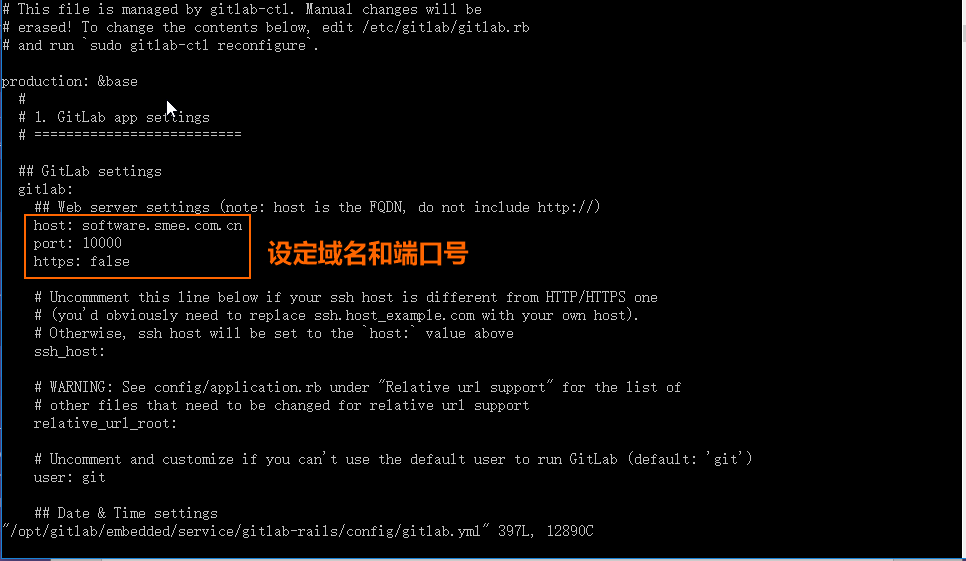


1. **配置gitlab**

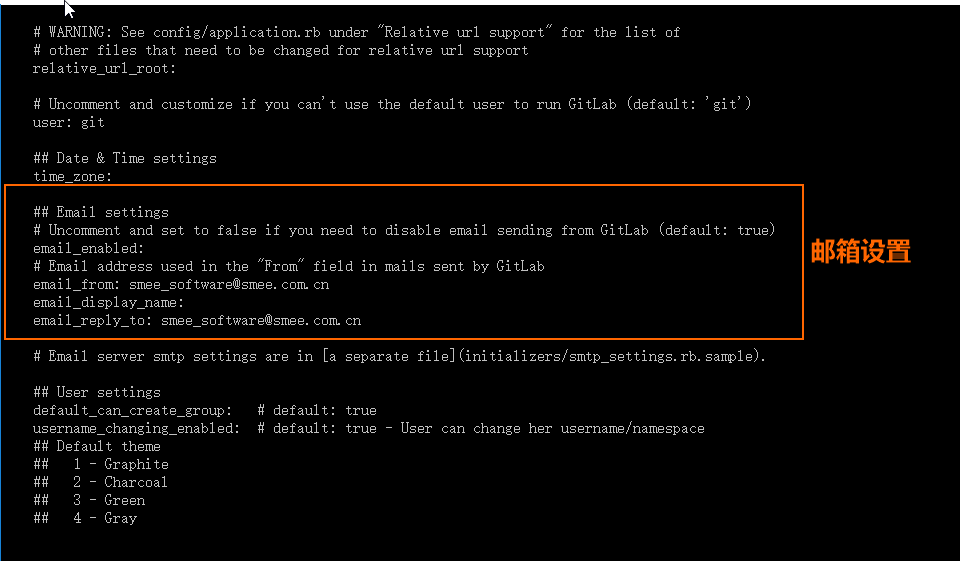
**首先找到配置文件**



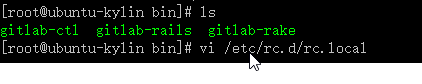
**其次，修改配置文件中的域名和端口号**



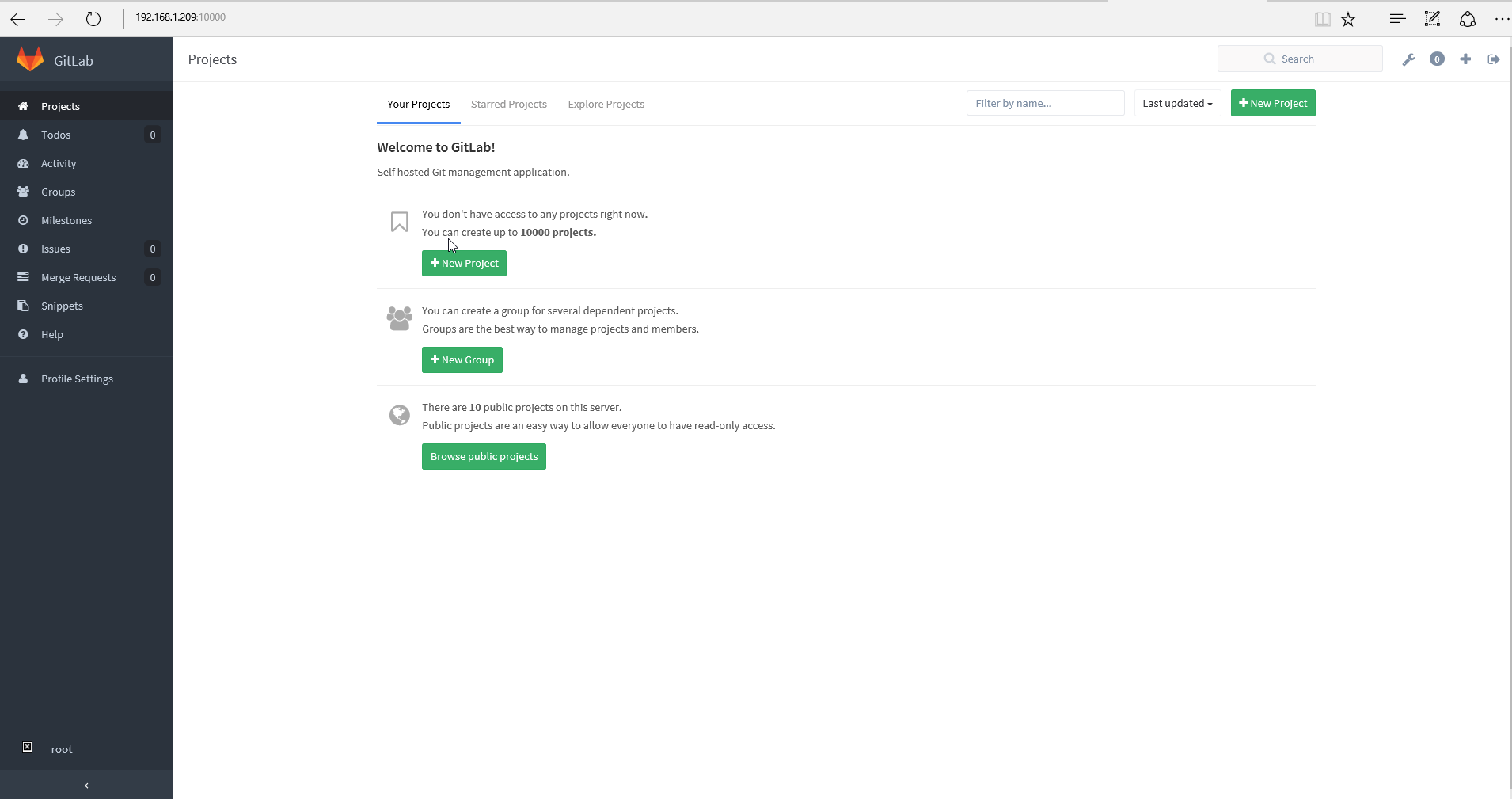
**邮箱设置**



1. **启动gitlab**



1. **登录gitlab，url=http://192.168.1.209:10000/**



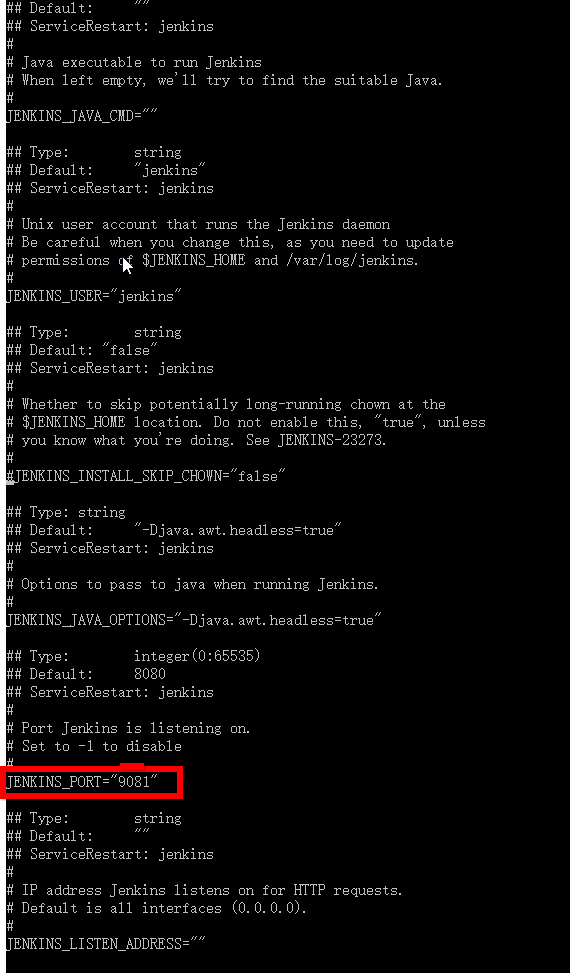
1. **Jenkins的安装及Jenkins + Gitlab环境搭建**
2. **从url =** [**http://pkg.jenkins-ci.org/redhat-stable/**](http://pkg.jenkins-ci.org/redhat-stable/) **之中下载最新版本的jenkins-1.651.2-1.1.noarch.rpm，并上传至root账户的soft文件夹中。**
3. **打开终端，输入 以下命令进行jenkins的安装**

**$ cd soft**

**$ rpm -ivh jenkins-1.651.2-1.1.noarch.rpm**

1. **配置端口号：$ vi /etc/sysconfig/jenkins**

**找到下图标注的区域，进行端口号的配置。**

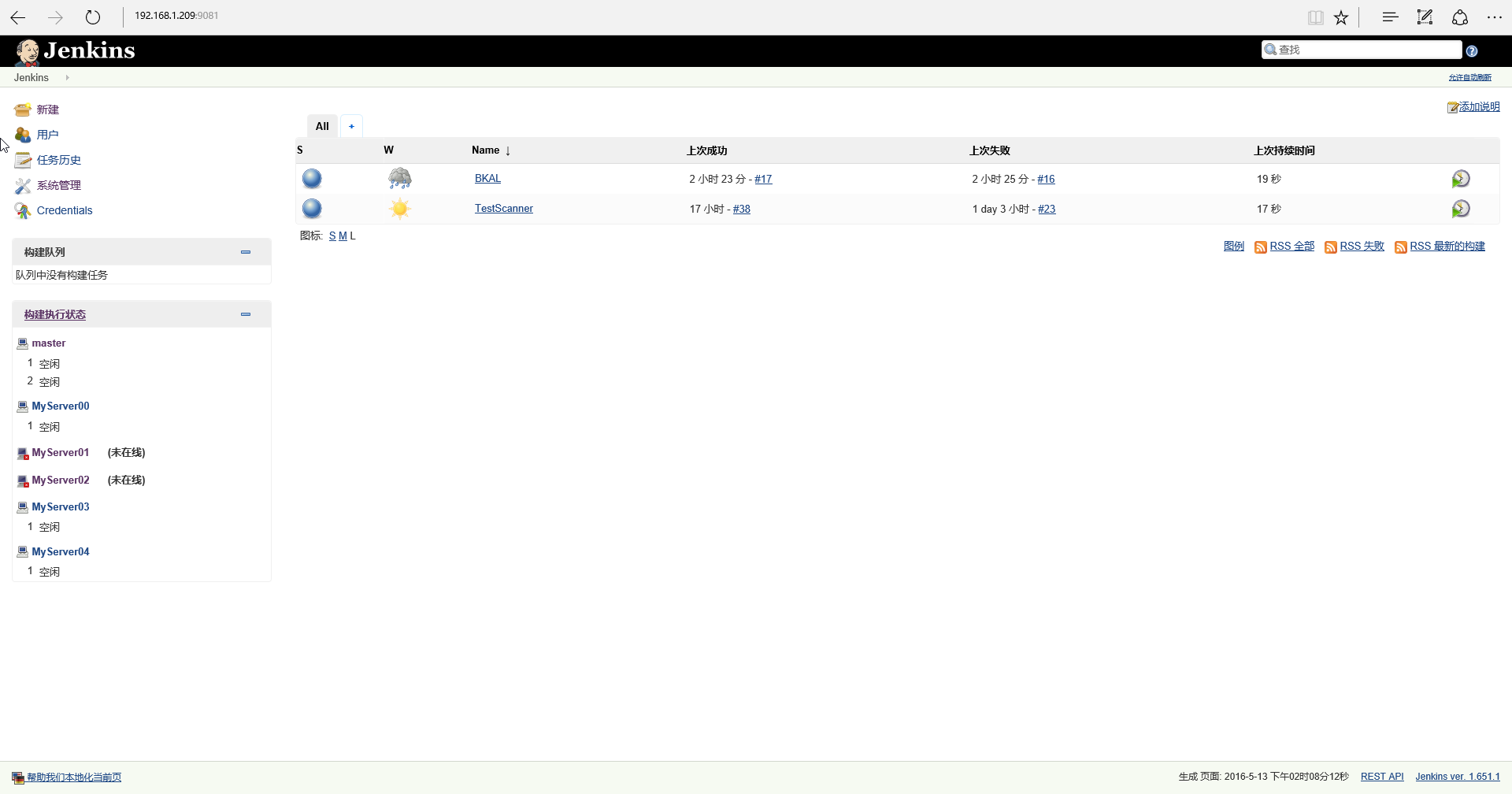


1. **设置开机自启动**

**$ sudo service jenkins start**

**$ sudo chkconfig jenkins on**

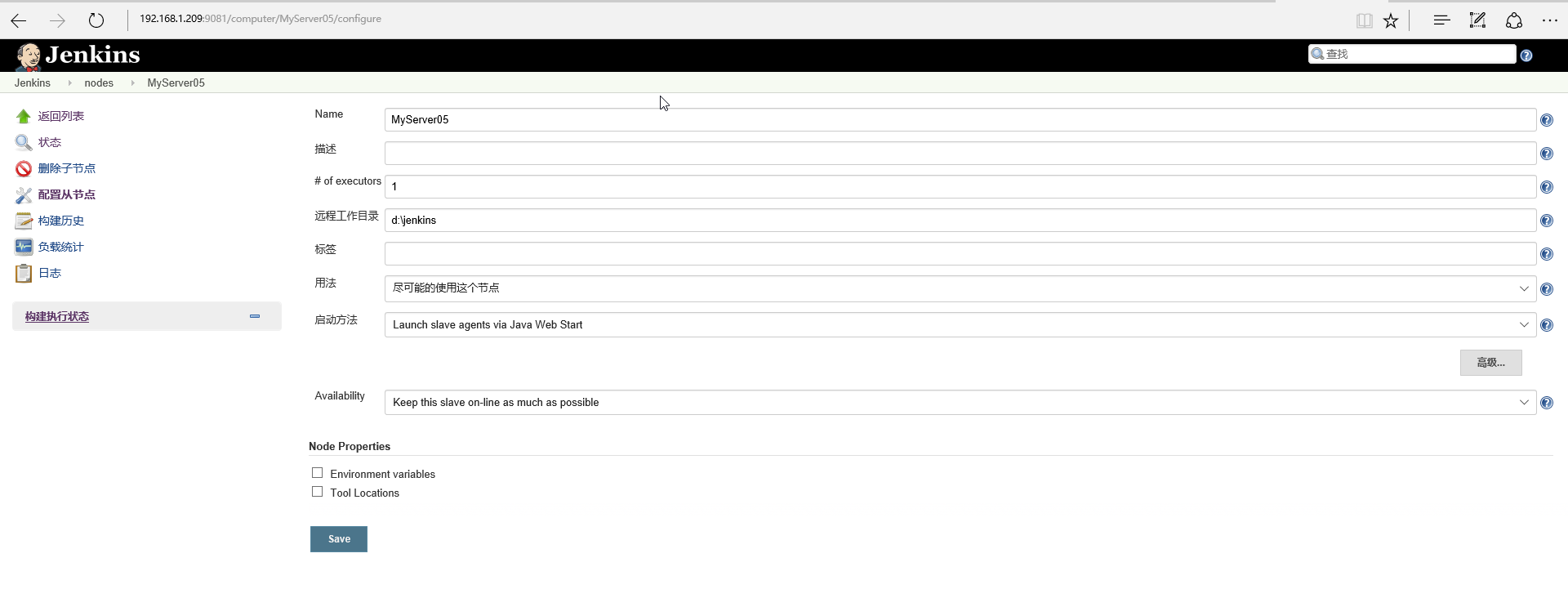
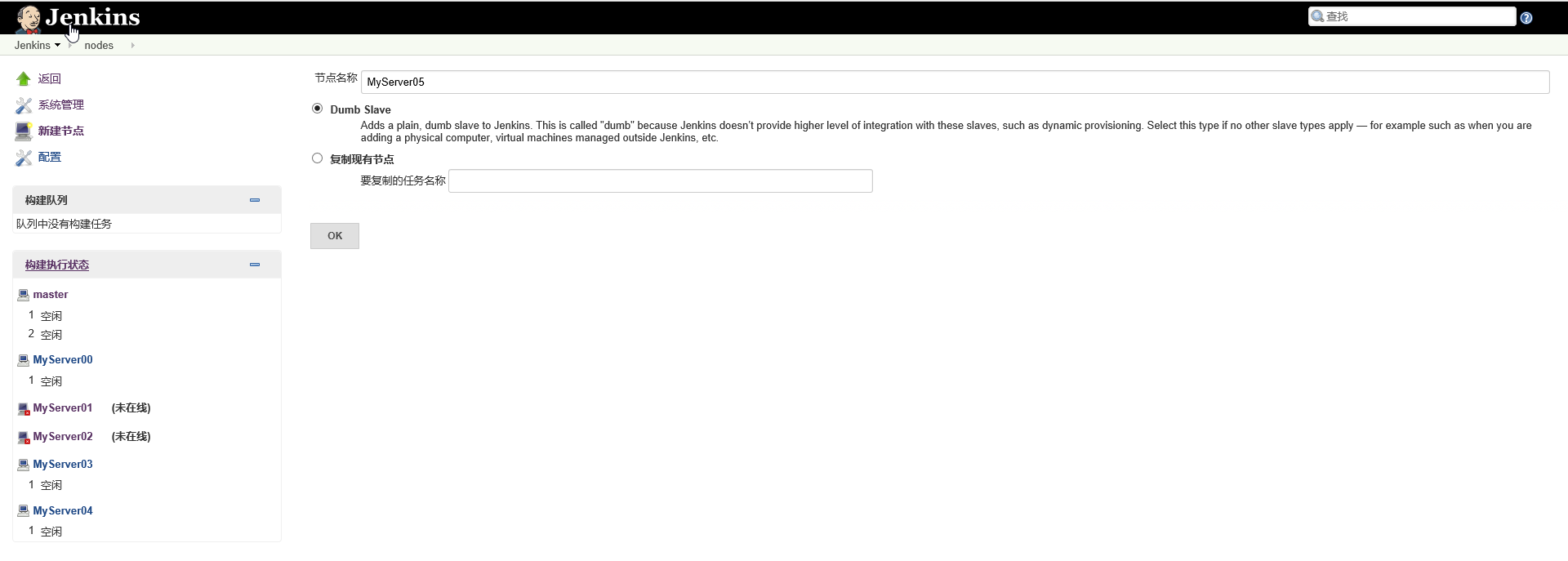
1. **登录jenkins, url = 错误！超链接引用无效。 address>:9081/**



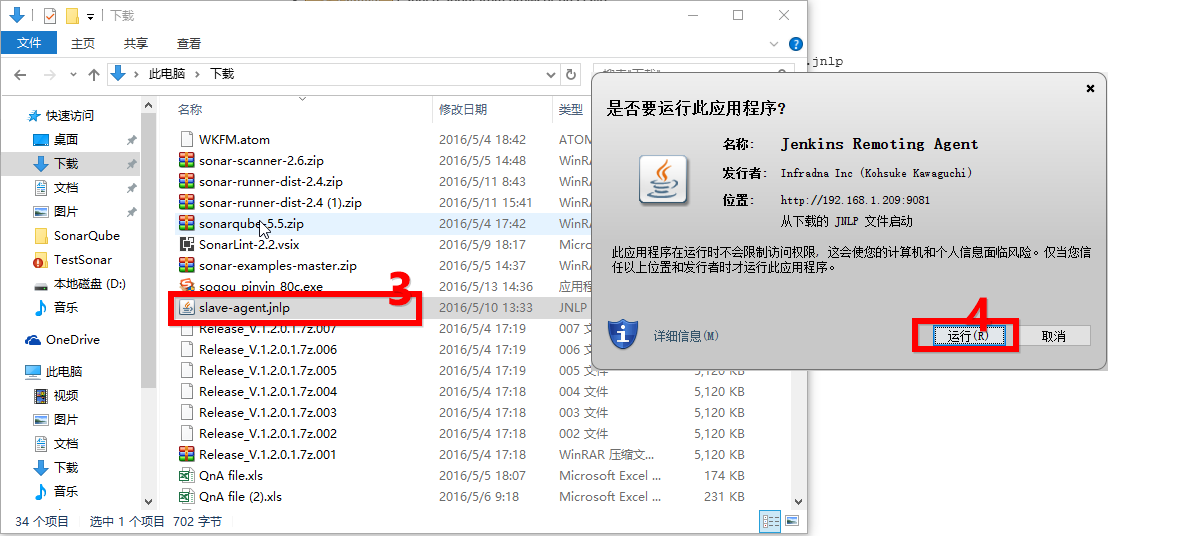
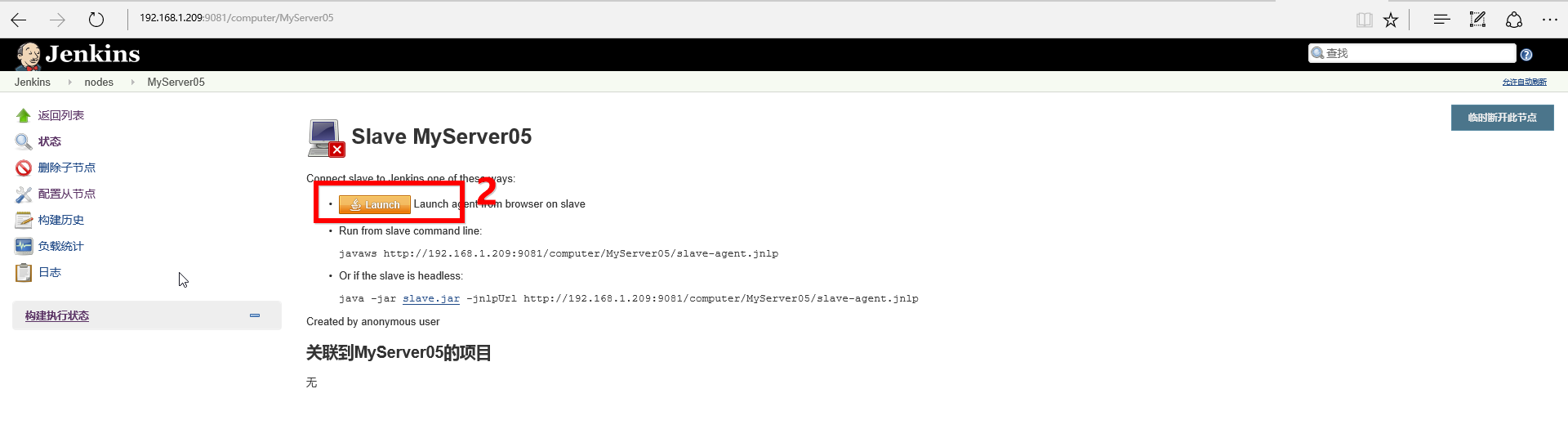
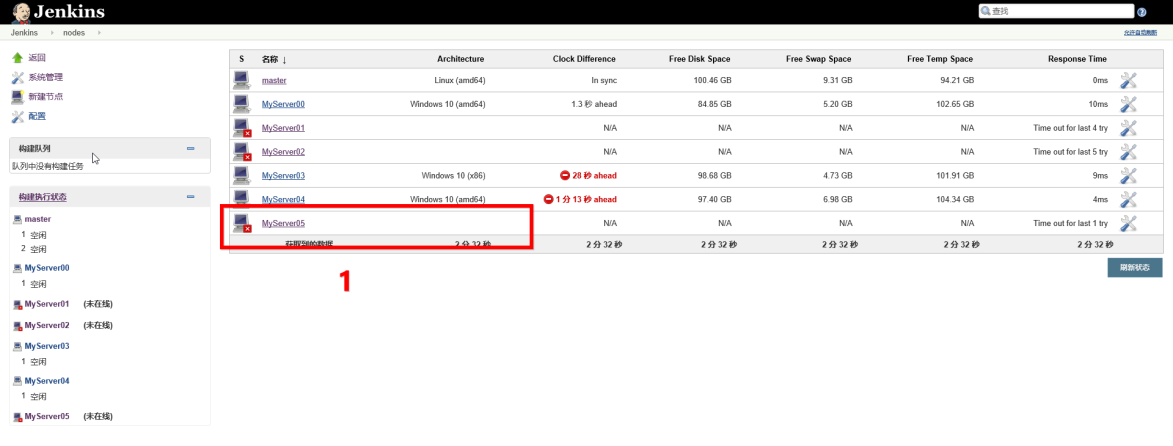
1. **集成gitlab + jenkins，安装所需的插件，方法如下：**

**点击“系统管理”—>“管理插件”—>“可选插件”，勾选“Email Extension Plugin”/“Git client plugin”/ “Git plugin”/ "Git Tag Message Plugin"/ "Gitlab Hook Plugin"/"GitLab Plugin"/"MsBuild Plugin"/"SonarQube Plugin",直接安装.**

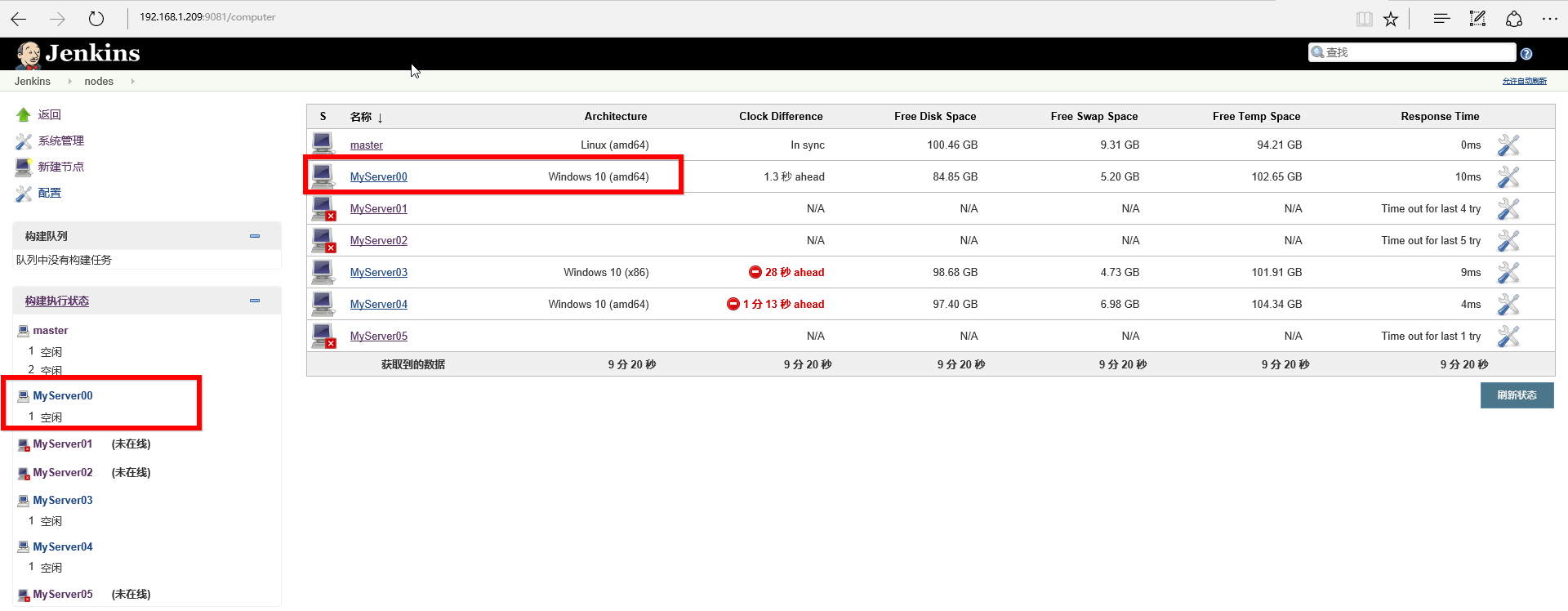
1. **创建slave节点:"系统管理"->"管理节点"->"新建节点".**



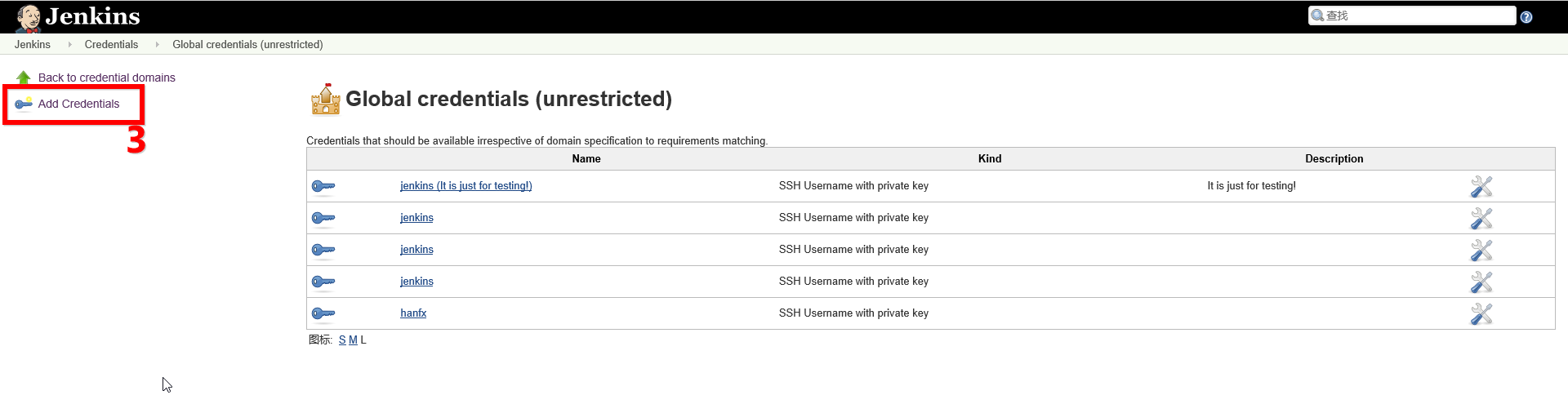
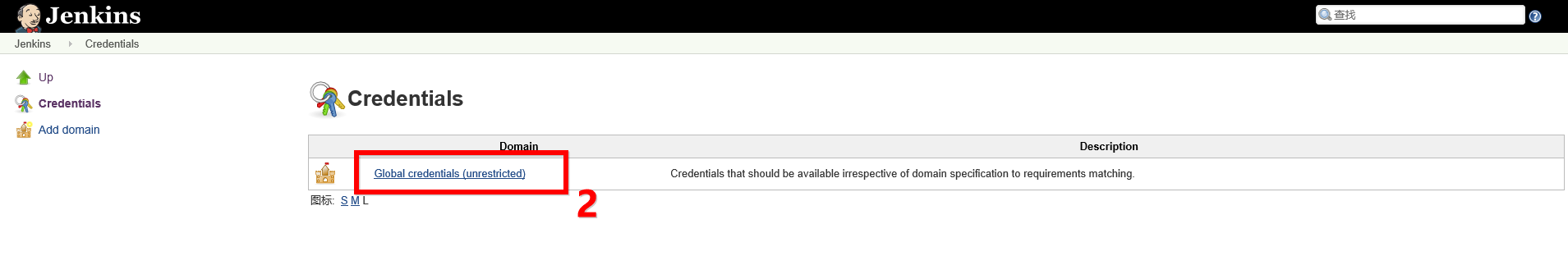
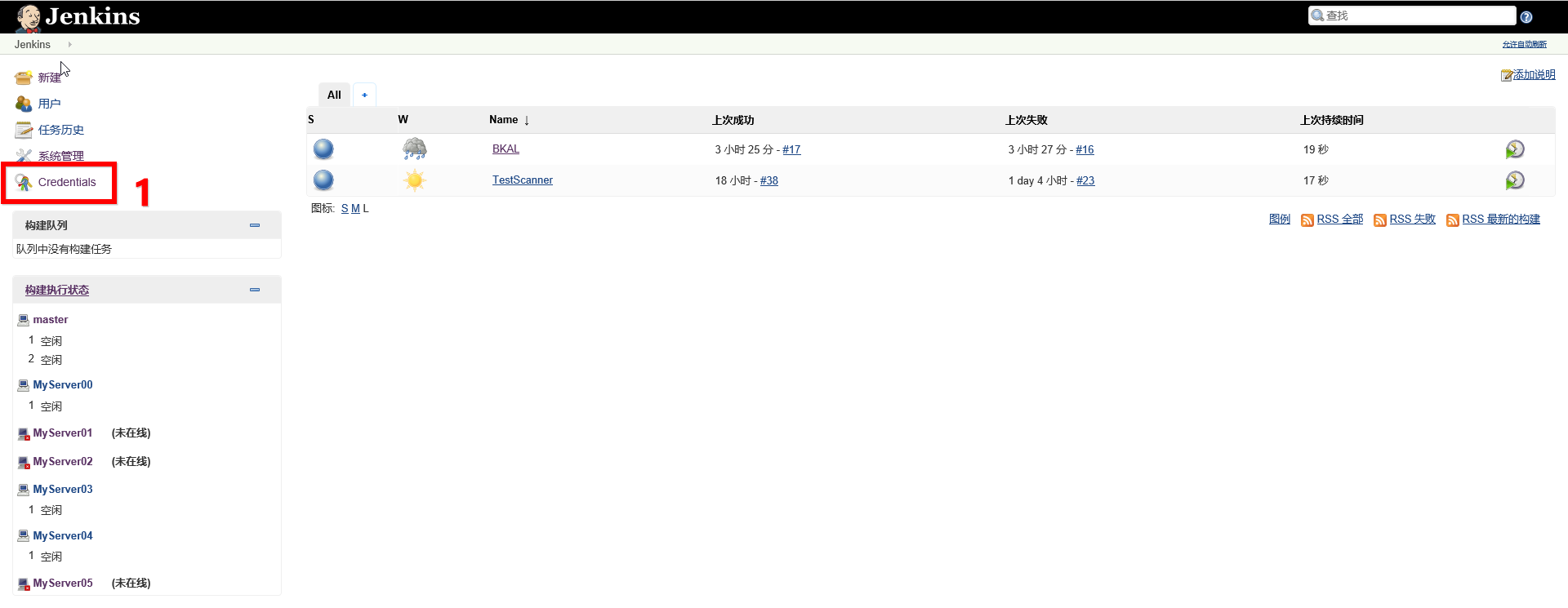
1. **从slave机器登录子节点:登录jenkins网页,选择相应的子节点,这里以子节点Myserver05为例,**



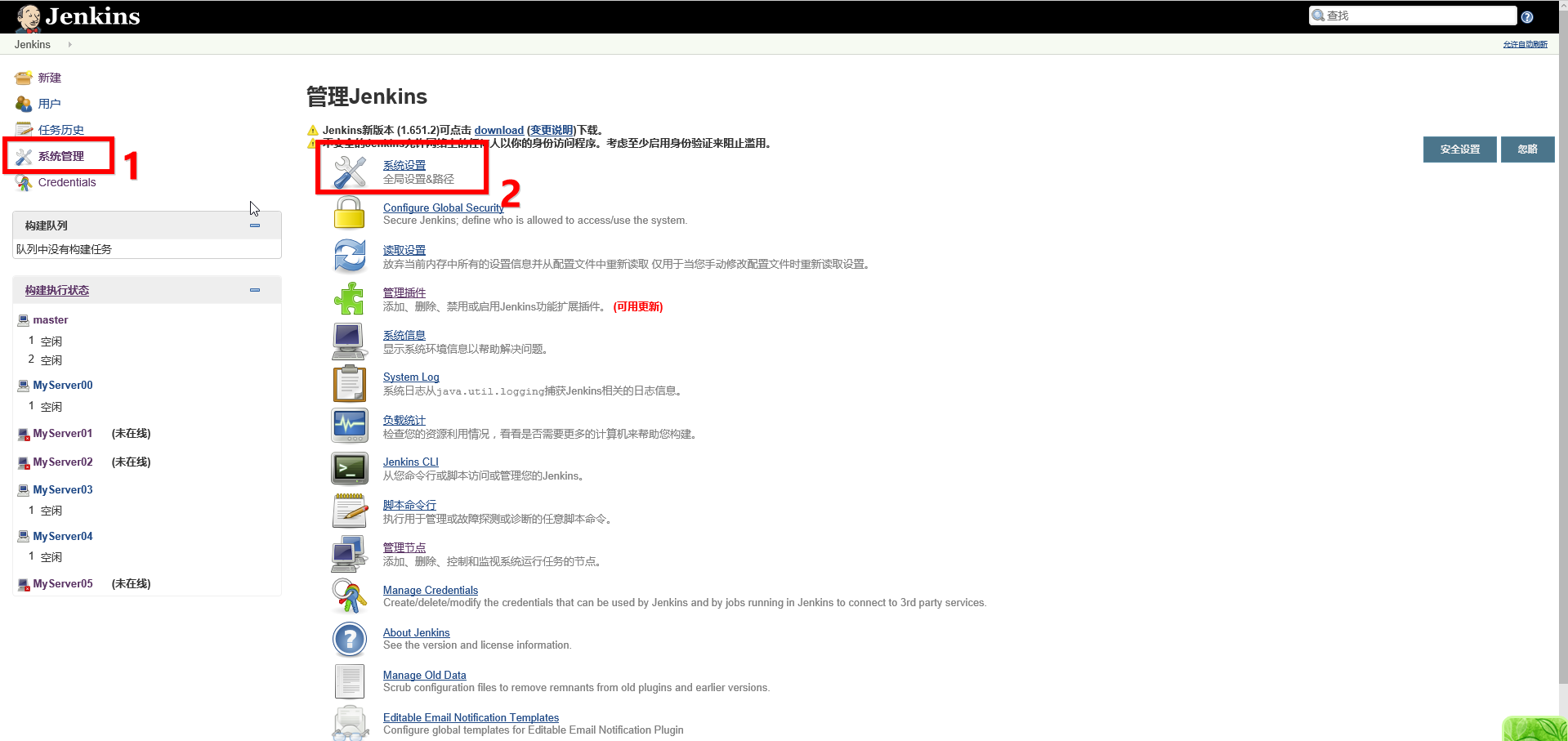
**登录进去后的状态如下:**



1. **创建Credentials,步骤如下:**



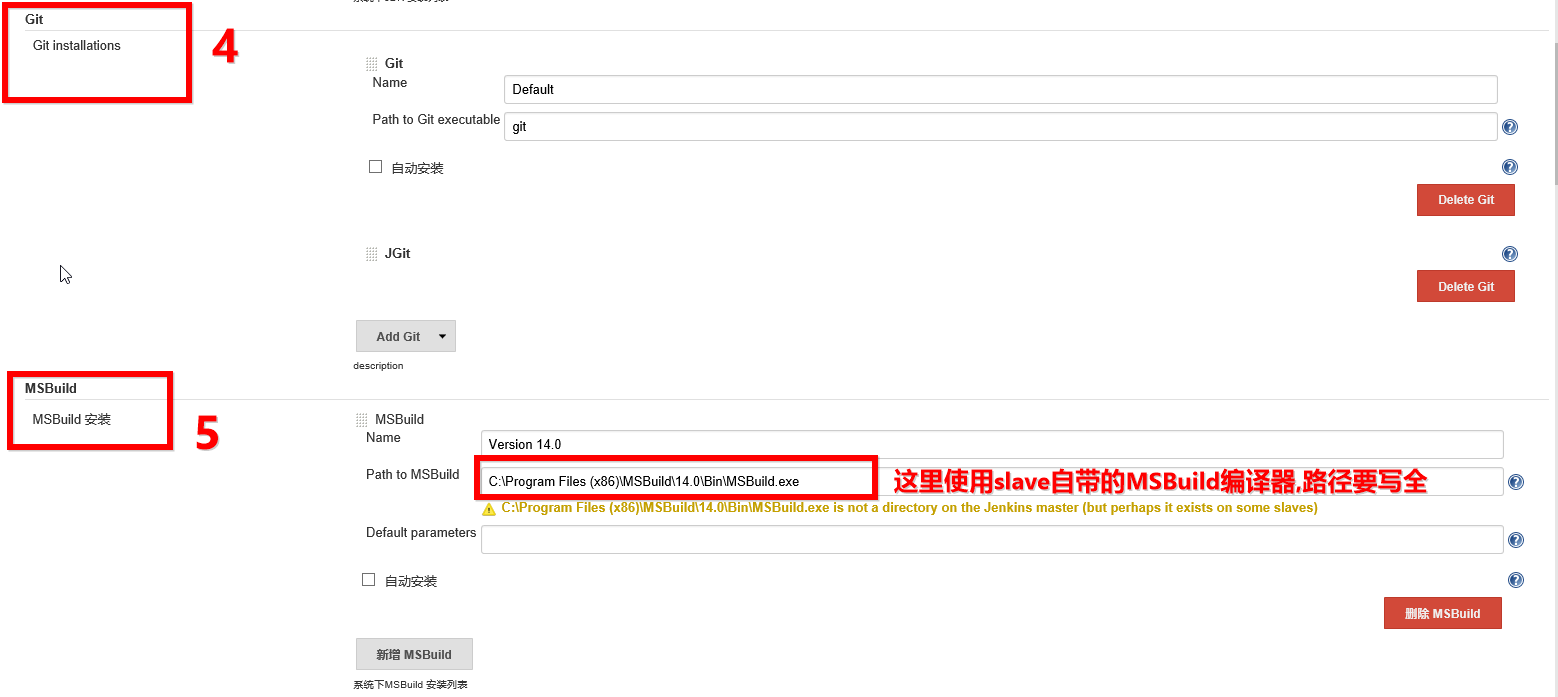
1. **系统设置**



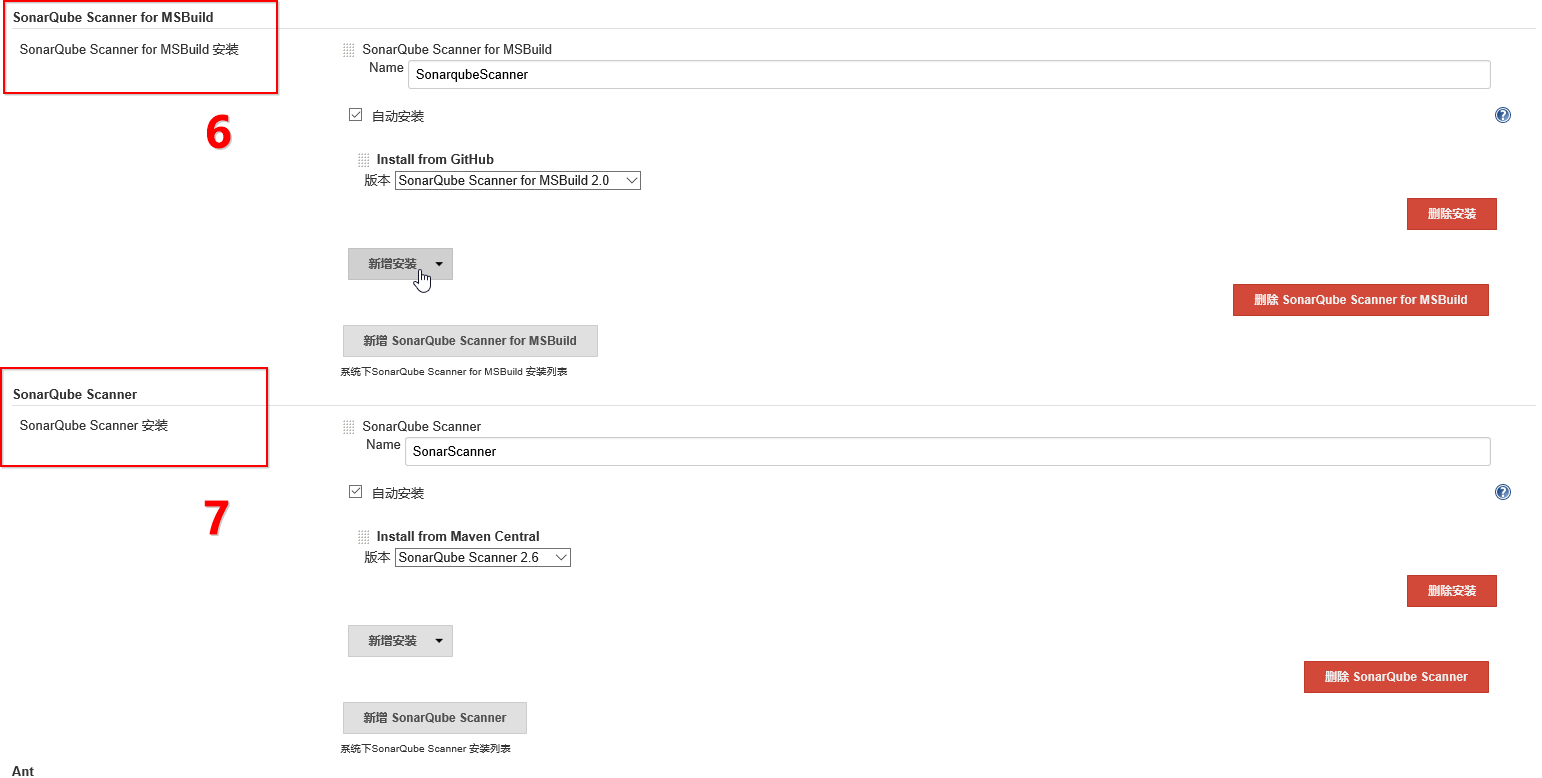
**添加SonarQube Servers**



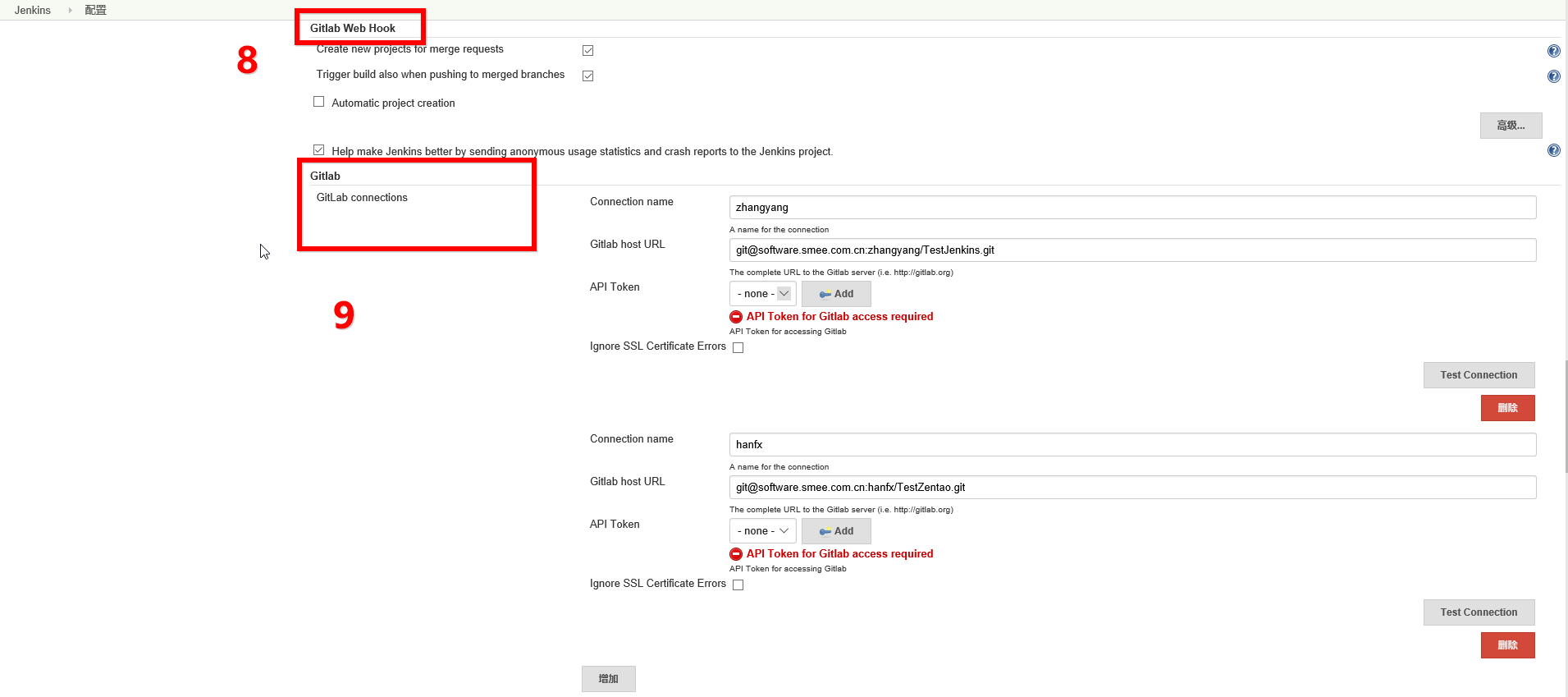
**添加Git和MSBuild**



**添加SonarQube Scanner for MSBuild 和 SonarQube Scanner**



**配置Gitlab Web Hook/Gitlab**



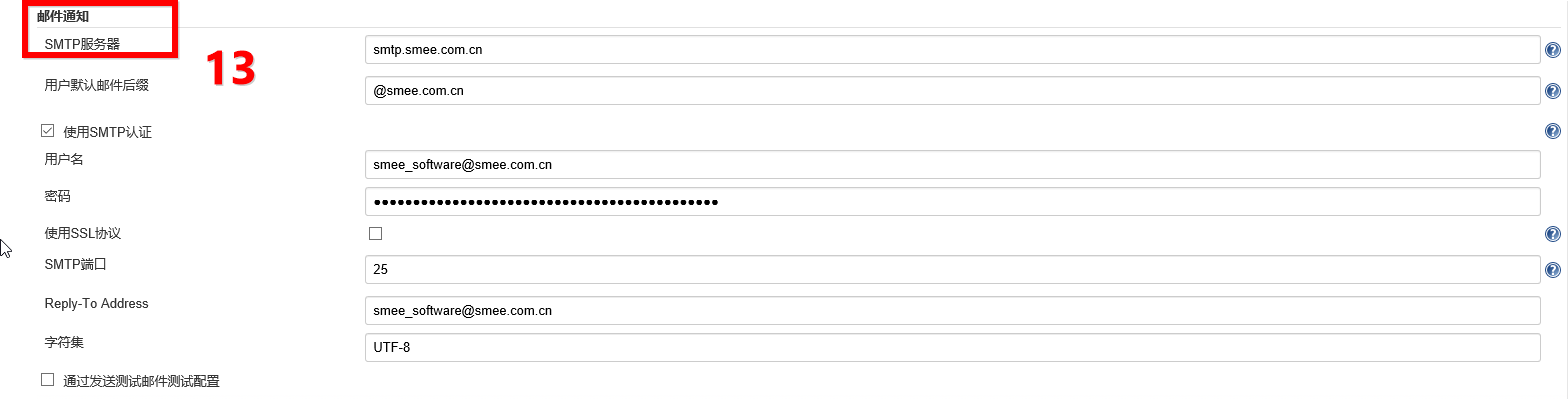
**配置Jenkins Location**



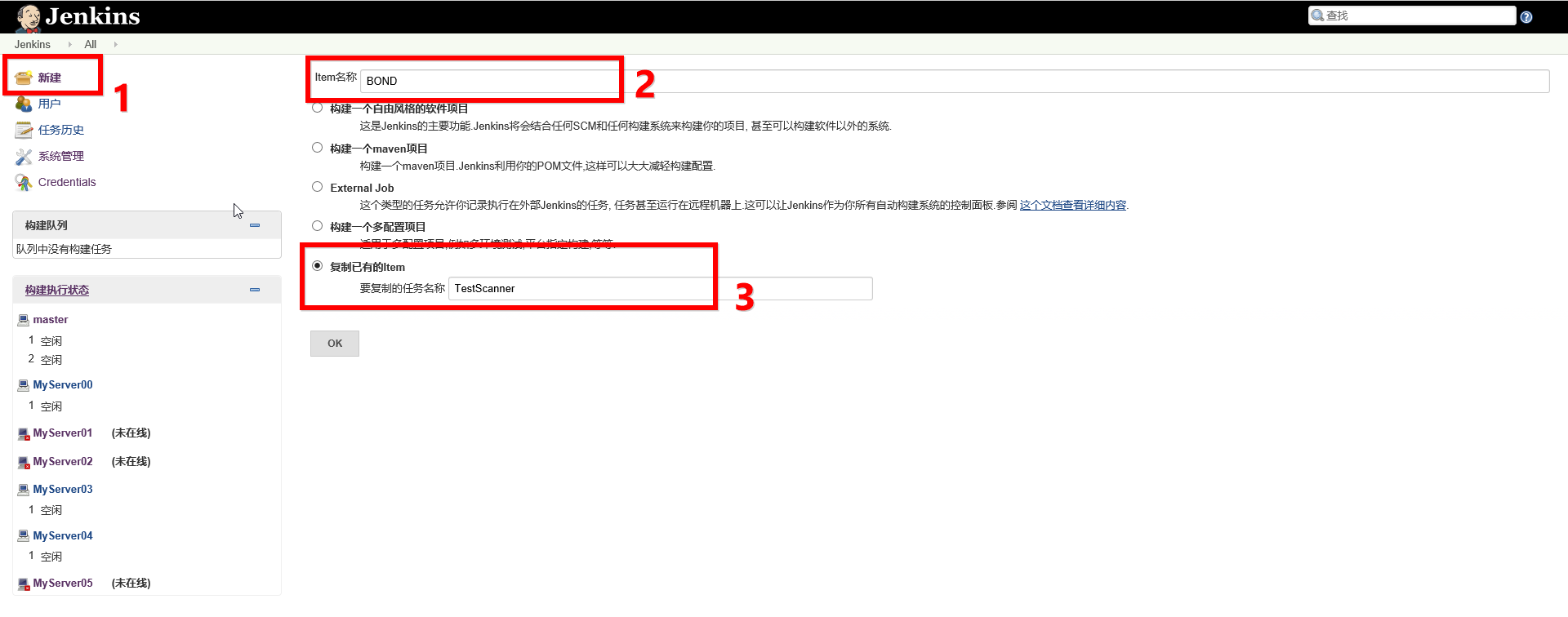
**配置Gitlab Notifer**



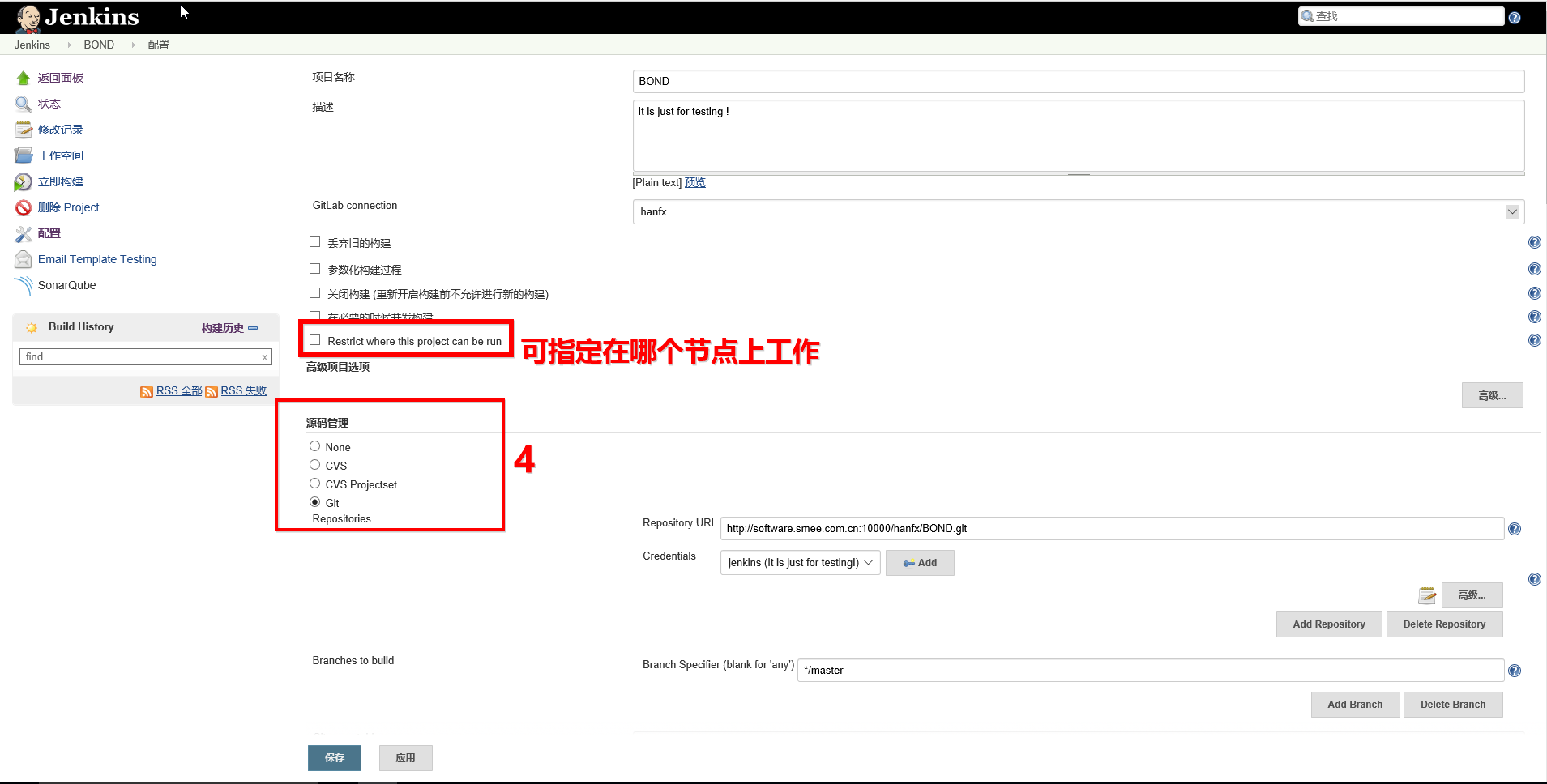
**配置邮件通知**



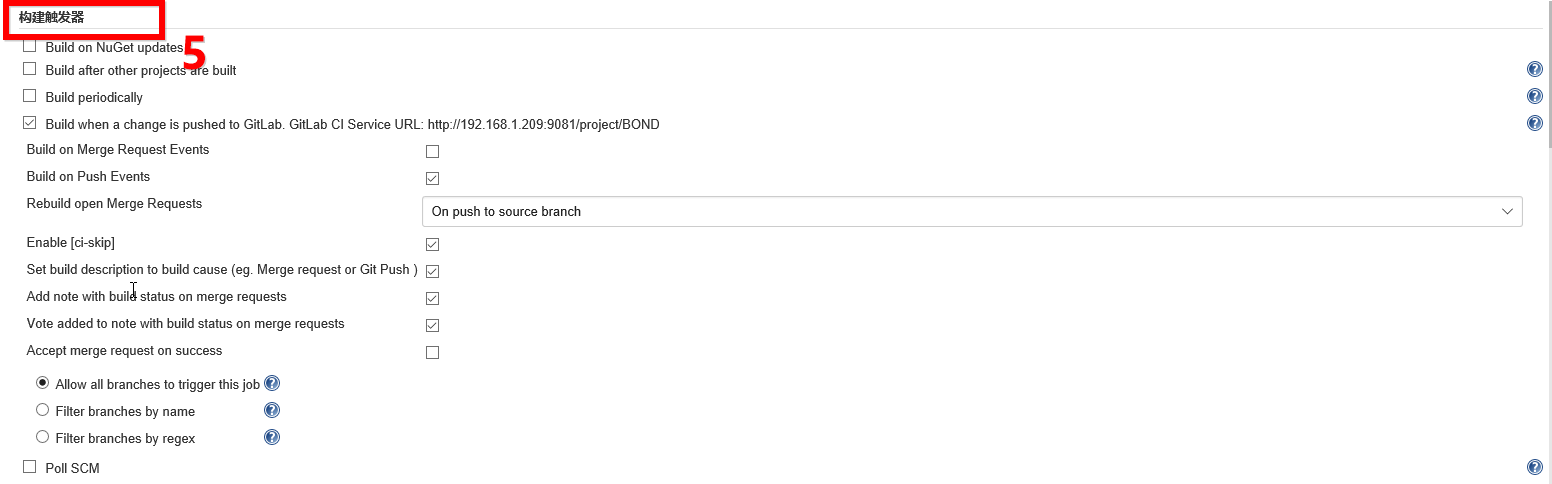
1. **创建Job**



**配置源码管理**



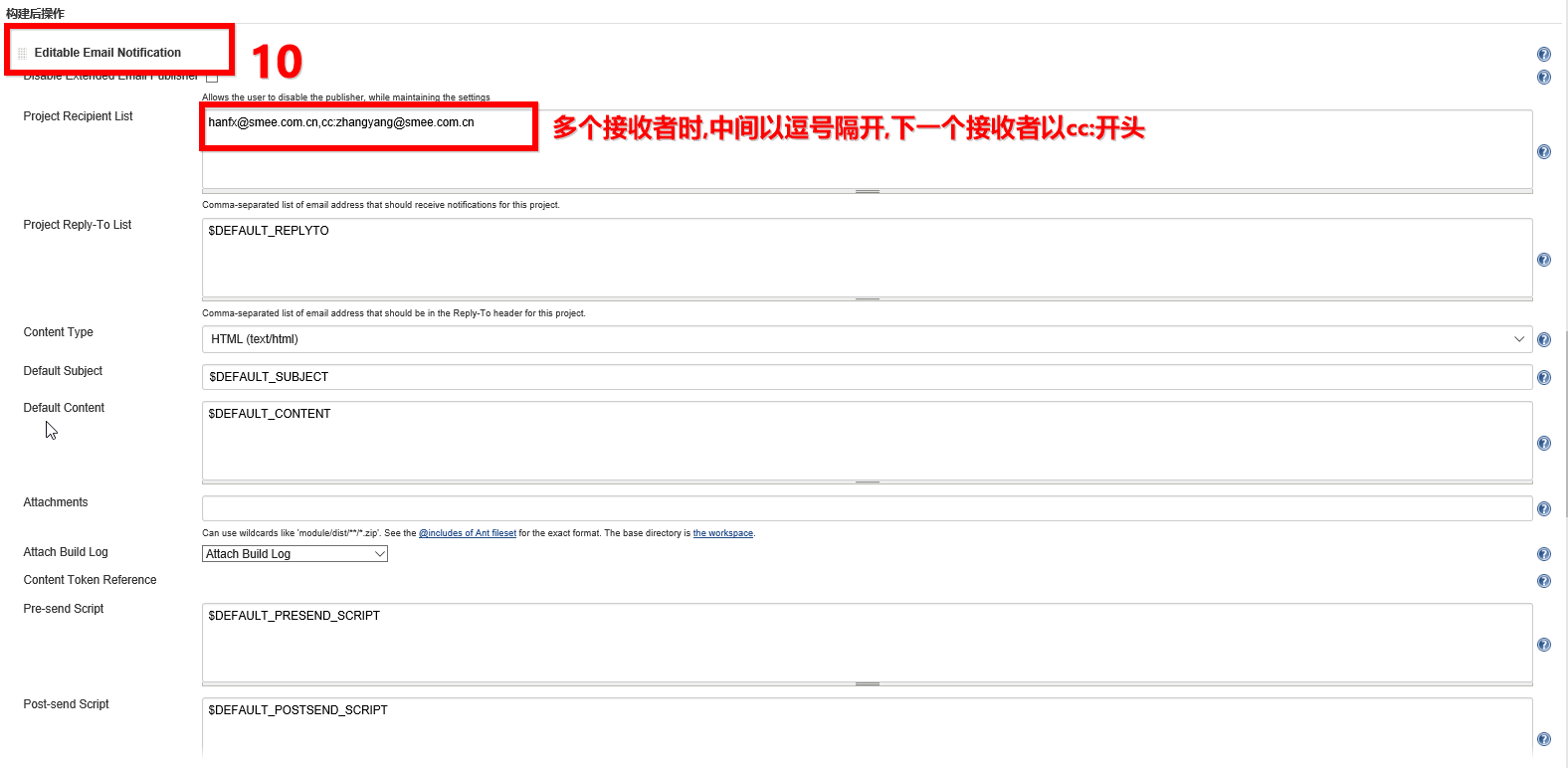
**构建触发器**



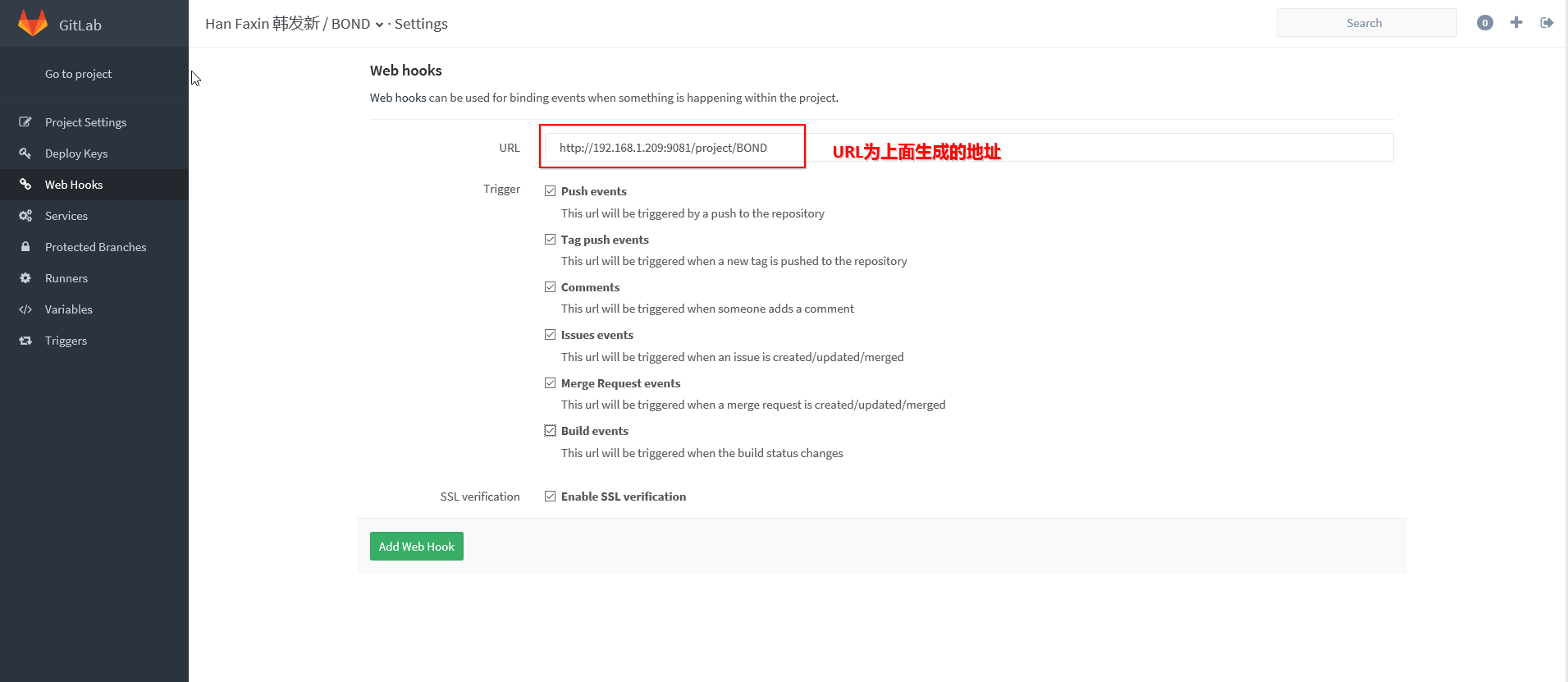
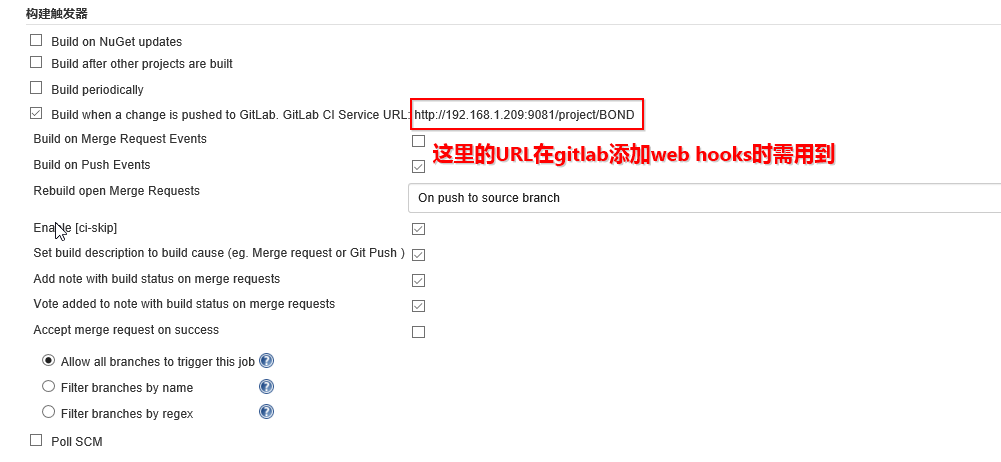
**构建:SonarQube Scanner for MSBuild – Begin Analysis->Build a Visual Studio project or solution using MSBuild->SonarQube Scanner for MSBuild – End Analysis->Execute SonarQube Scanner**



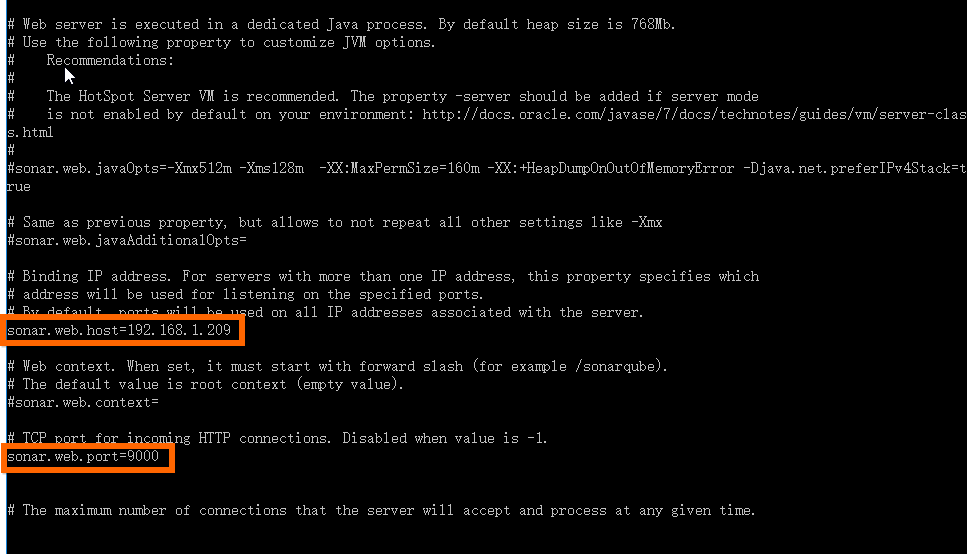
**构建后操作:这里构建了邮件通知服务**



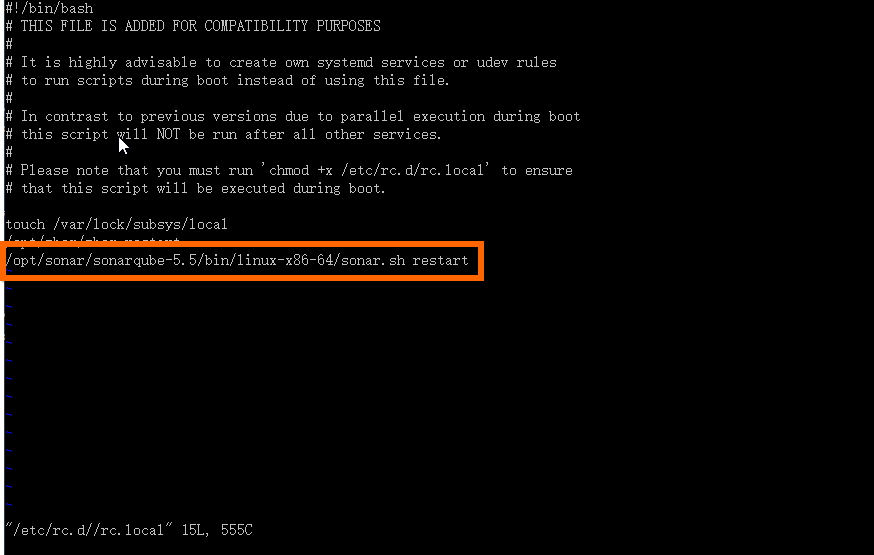
1. **Gitlab项目中web hooks的添加:**



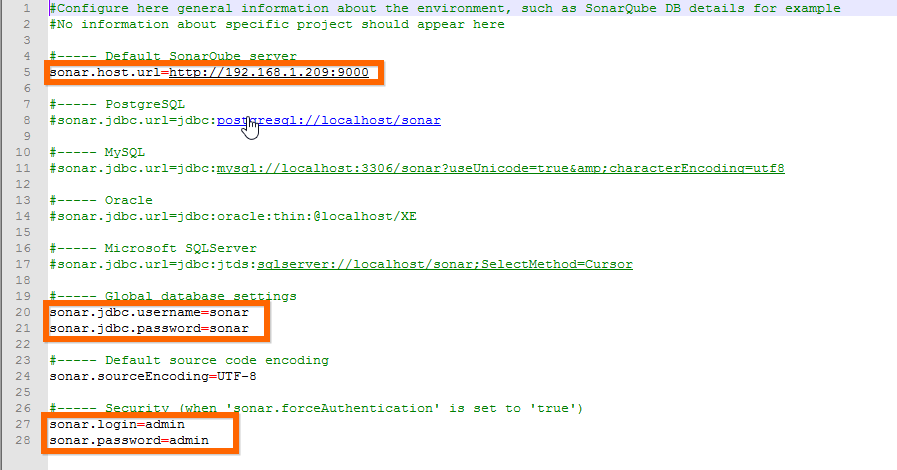
1. **Nuget 服务器搭建及Jenkins中Nuget的集成**
2. **Sonarqube的安装及环境搭建**
3. **从url=https://sonarsource.bintray.com/Distribution/sonarqube/sonarqube-5.5.zip下载sonarqube的安装包，存放到指定的路径下，这里我们解压到/opt/sonar/下。**
4. **修改/opt/sonar/sonarqube-5.5/conf/sonar.properties配置文件，设定sonar.web.host 和 sonar.web.port 。**



1. **设定开机自启动：在/etc/rc.d/rc.local添加启动项**

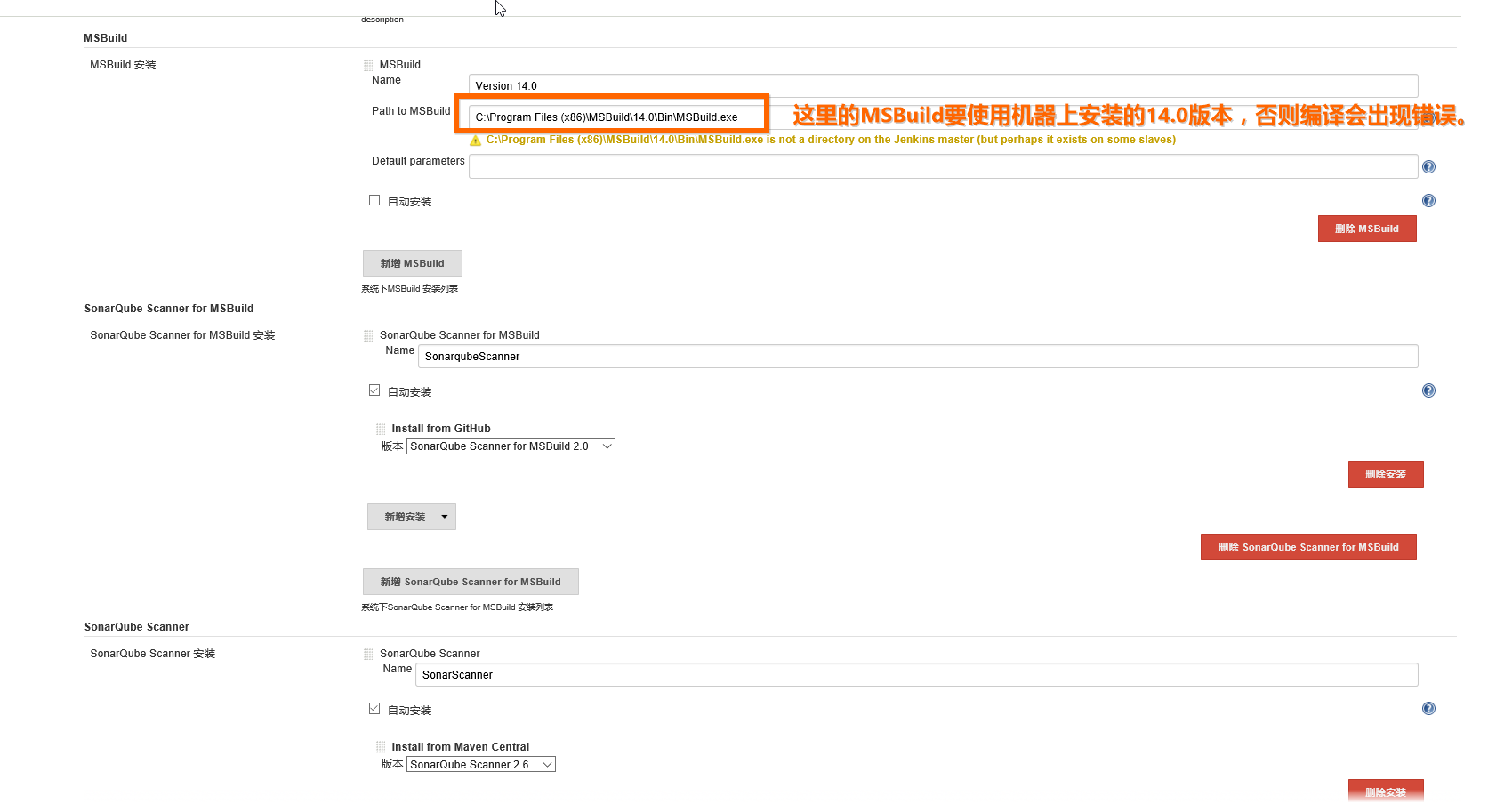


1. **从url=https://github.com/SonarSource-VisualStudio/sonar-msbuild-runner/releases/download/2.0/MSBuild.SonarQube.Runner-2.0.zip下载MSBuild.SonarQube.Runner的安装包解压到windows工作目录D:\jenkins\SonarQube\，并修改SonarQube.Analysis.xml配置文件中的sonar.host.url为远程服务器IP。**
2. **下载sonarqube Runner，解压到D:\jenkins\SonarQube\下，修改sonar runner的数据库配置，打开文件D:\jenkins\SonarQube\sonar-runner-2.4\conf\** **sonar-runner.properties 。根据数据库使用情况进行取消相关的注释即可，同时需要添加数据库用户名和密码信息，本环境的用户名和密码信息如图：**



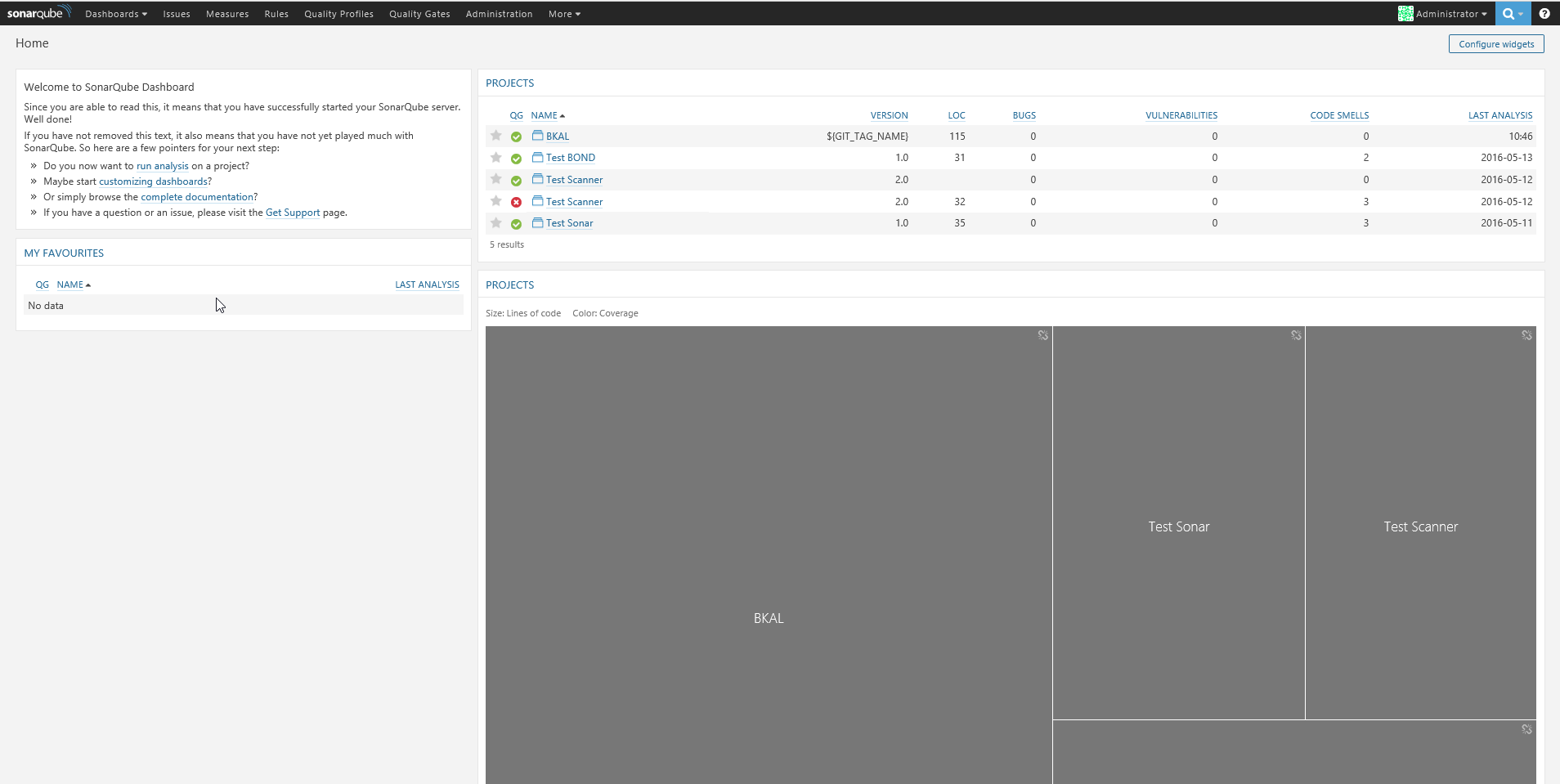
1. **在jenkins中构建sonarqube:**

**首先在jenkins的系统管理配置中添加安装MSBuild、SonarQube Scanner for MSBuild以及SonarQube Scanner。如下图：**

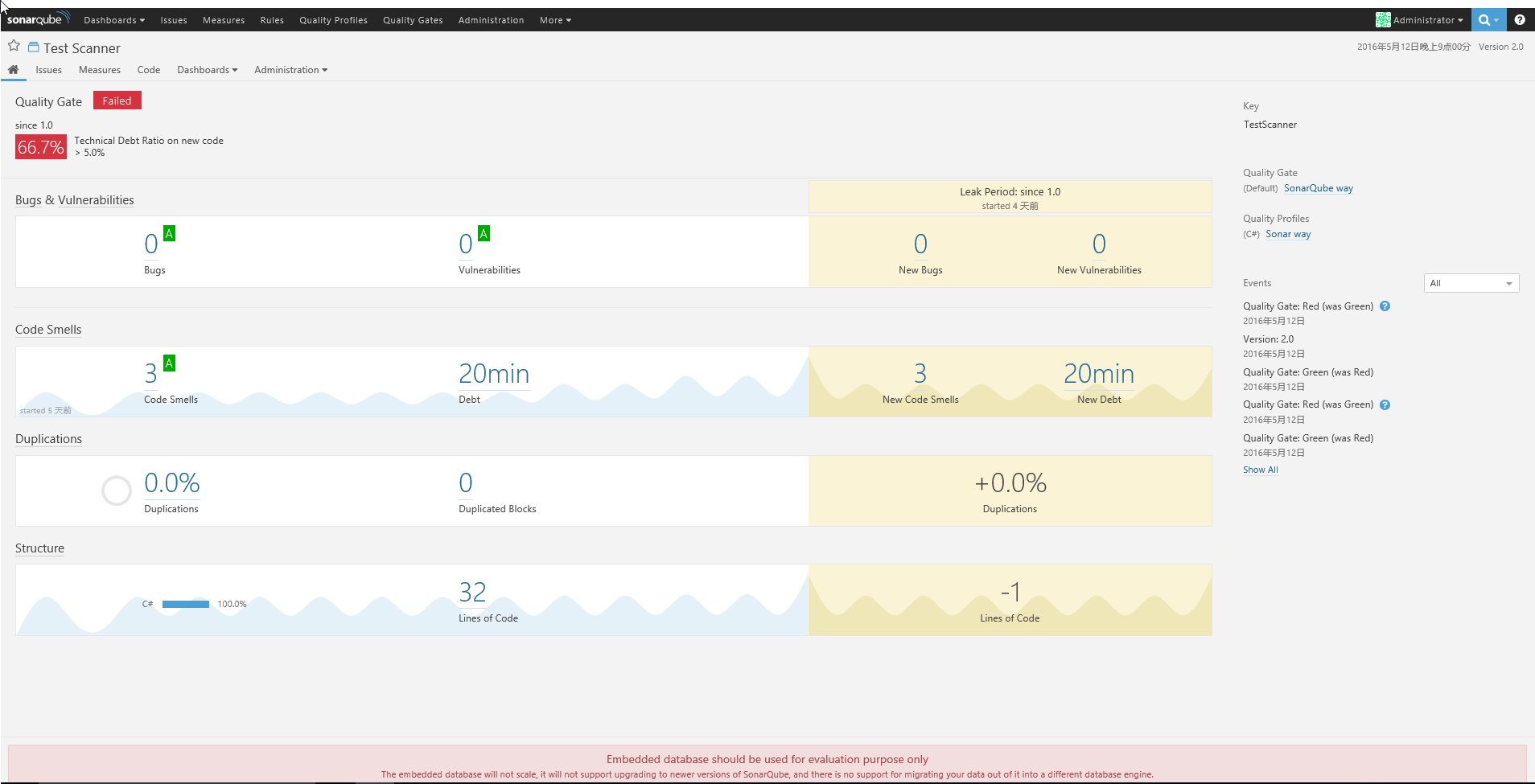


**其次，在对应的Job配置中，进行任务的构建，具体构建已在Jenkins的配置中描述。**

1. **运行jenkins中的任务，结束后会将信息发送给sonarqube服务端，显示如下：**



**进入对应的job可查看具体的信息：**

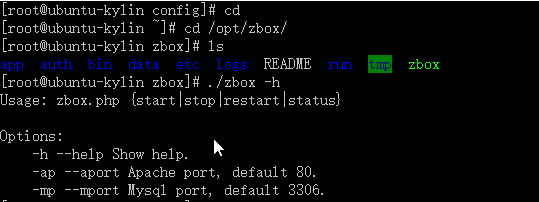


1. **ZenTao的安装**
2. **从禅道官网下载ZenTaoPMS.8.1.stable.zbox\_64.tar.gz安装包，并上传服务器中；**
3. **解压禅道安装包到/opt/下，修改apache和mysql的端口:**

**$ cd /opt/zbox/**

**$ ./zbox -ap 20001 ------设置apache端口**

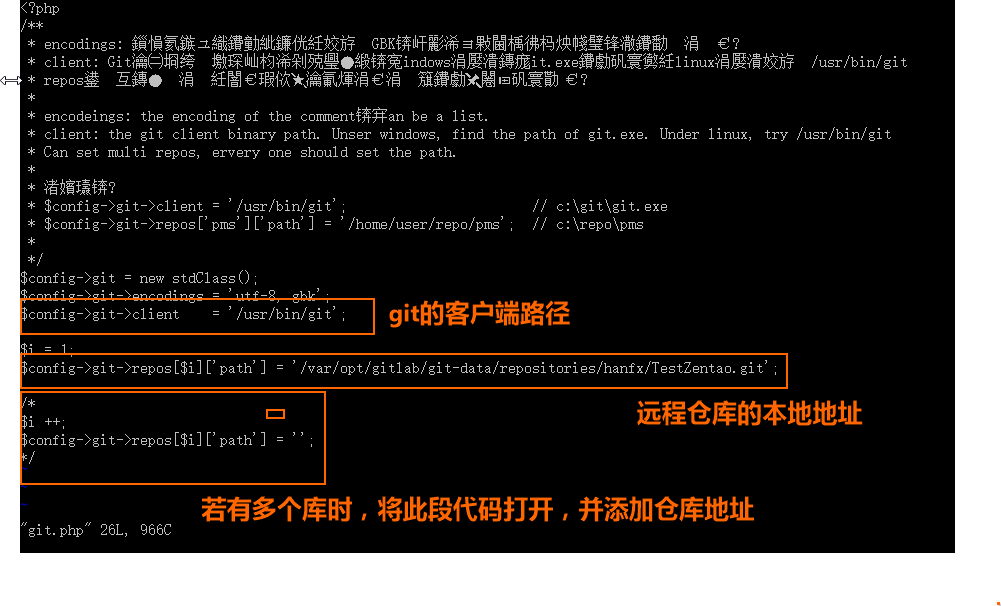
**$ ./zbox restart -------启动禅道**



1. **集成ZenTao + Git**
2. **配置禅道关于git的参数**

**首先，找到位于/opt/zbox/app/zentaopro/module/git/下的config.php文件，复制该文件到/opt/zbox/app/zentaopro/module/git/ext/config/下，并命名为git.php；**

**其次，修改git.php文件:**

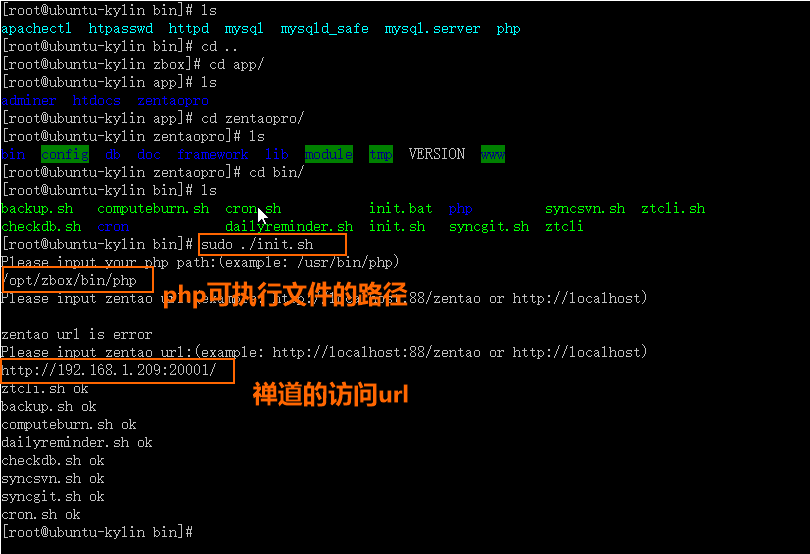


1. **代码提交注释格式：**

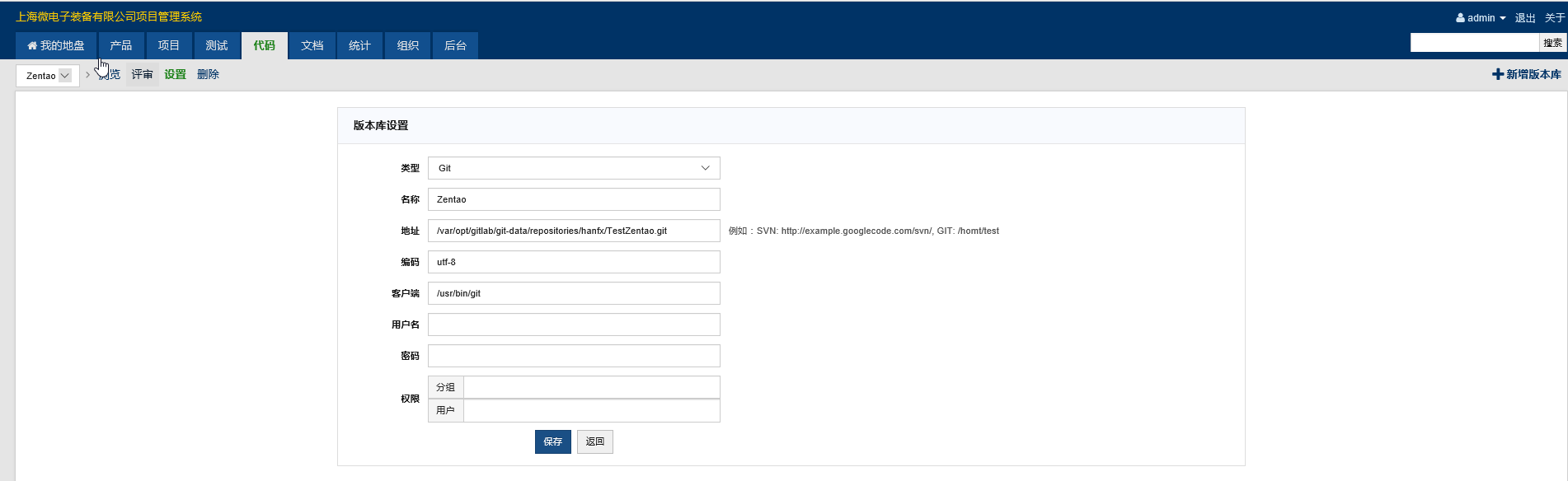
**开发者在提交代码到gitlab的时候，需要在备注里面注明此次修改相关的需求，或者任务，或者bug的id。比如下面的格式：**

**bug#123, 234, 1234, 也可以是bug:123, 234 1234, id列表之间，用逗号和空格都可以。**

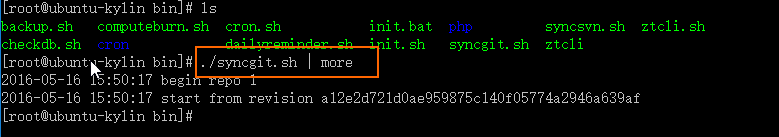
1. **初始化管理脚本：**



1. **网页端版本库设置**



1. **执行git同步命令：**



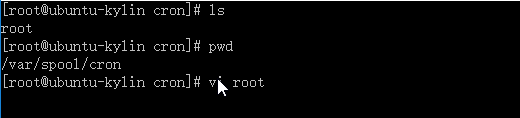
1. **部署定时任务：**

**网页端：**



**服务器上：**

**使用crontab创建一个crontab文件，创建好的文件在/var/spool/cron/下，在crontab文件中添加定时刷新的内容：**



**这里的意思是每5分钟执行一次syncgit.sh脚本。**

1. **一切OK后的网页端情况如下图：**

