Jenkins部署

1. 安装jdk
2. 将jdk-8u111-linux-x64.tar.gz放到/opt目录下，解压tar -zxvf jdk-8u111-linux-x64.tar.gz
3. 在/etc/profile中配置jdk环境变量

 #配置jdk

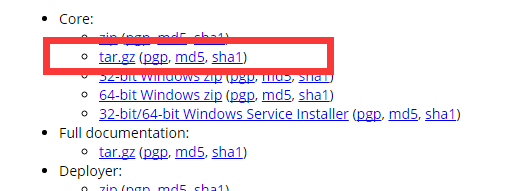
 JAVA\_HOME=/opt/jdk1.8.0\_111

 JRE\_HOME=$JAVA\_HOME/jre

 PATH=$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin

 CLASSPATH=:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar:$JRE\_HOME/lib/dt.jar

 export JAVA\_HOME JRE\_HOME PATH CLASSPATH

1. 通过javac 检验是否安装成功
2. 安装Tomcat
3. <http://tomcat.apache.org/download-70.cgi> 下载Tomcat
4. 解压Tomcat：#tar –zxvf apache-tomcat-9.0.0.M17
5. 进入bin目录，启动Tomcat：#sh startup.sh
6. 安装Jenkins
7. https://jenkins.io/content/thank-you-downloading-windows-installer/  下载.war包
8. systemctl stop firewalld //关闭防火墙

systemctl disable firewalld //防止防火墙开机启动

1. 将下载的安装包放在tomcat的webapps目录下，配置环境变量，设定Jenkins构建项目的目录：#vi /etc/profile

#配置Jenkins路径

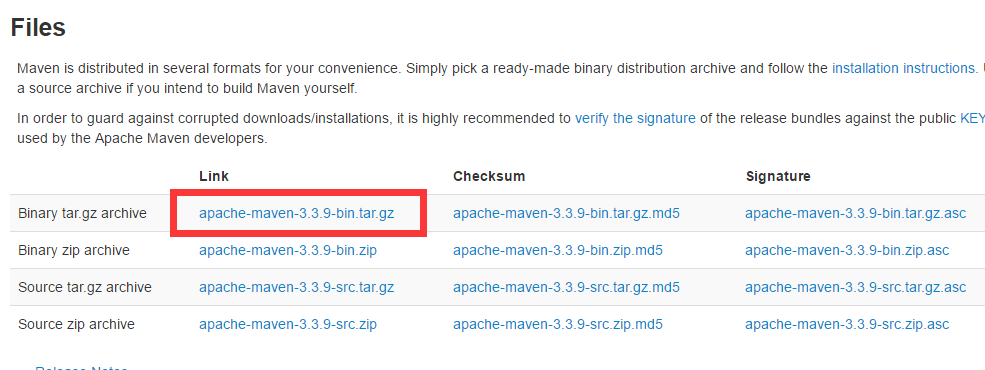
export JENKINS\_HOME=/home/jenkins

1. 建立一个文本并编辑：#vi start\_jenkins.sh（建立文本的好处是能够在后台运行，不至于关掉远程工具后程序停止运行）

#!/bin/bash

java -jar jenkins.war --httpPort=8082 &

1. 在浏览器中打开http://ip:8082
2. 要求你根据页面中的路径输入密码（/home/Jenkins/secrets/initialAdminPassword），也可以在shell界面复制，安装系统推荐的组件，并注册用户，进入Jenkins界面。
3. 安装maven
4. <https://maven.apache.org/download.cgi>下载



1. 将其放到/opt目录下并解压：#tar –zxvf apache-maven-3.3.9-bin.tar.gz
2. 配置maven环境变量：#vi /etc/profile

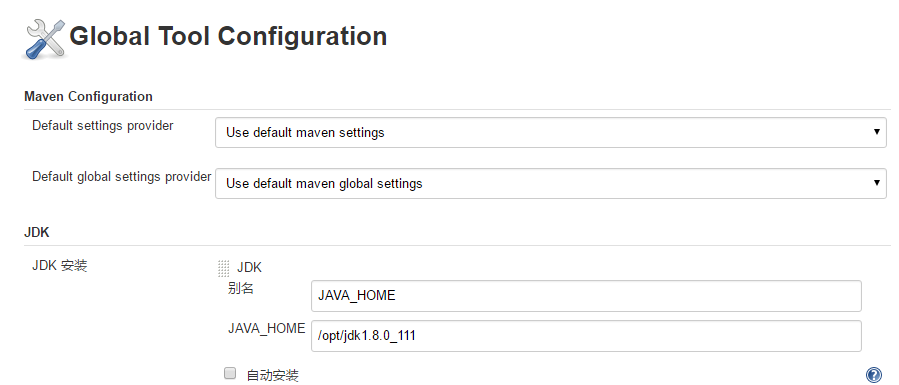
#配置maven

export MAVEN\_HOME=/opt/apache-maven-3.3.9

export PATH=$PATH:$MAVEN\_HOME/bin

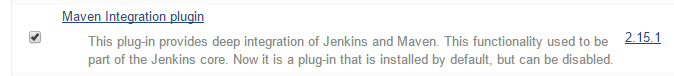
1. 进入“系统管理”，配置jdk和maven路径，步骤如下图：





1. 进入 “管理插件”中下载“Maven Integration plugin”插件，选择“直接安装”





1. 从“管理插件”中下载“Email Extension Plugin”



1. 配置邮件发送
2. 要想邮箱能够收到邮件，需开启smtp服务，同时会获得“授权码”，系统管理员会到该授权码作为认证密码（在下面会用到），其他接受者只需开启smtp即可。

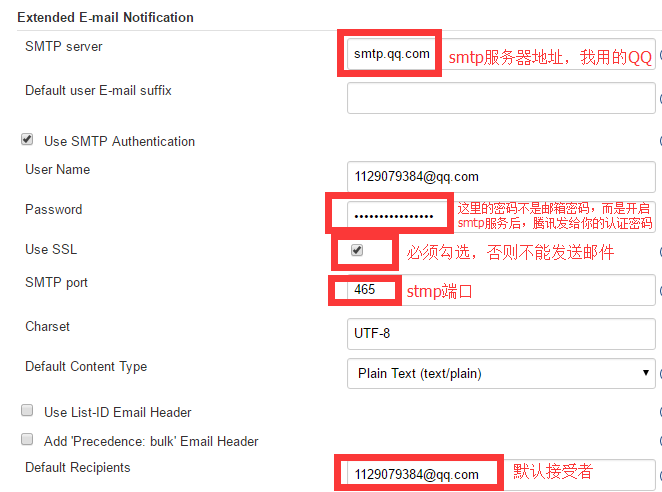




1. 进入“系统管理”，必须填写“系统管理员邮件地址”



1. 配置“Extended E-mail Notification”，这一步至关重要（**入过坑**）



1. 最后，可以测试邮件发送



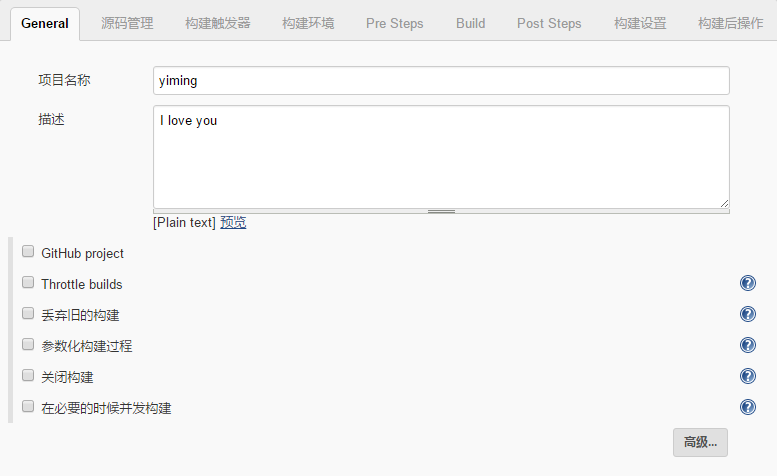
Jenkins项目构建

1. 新建一个maven项目（普通项目和选择“构建一个自由风格的软件项目”），随便取个名字，选择“构建一个maven项目”

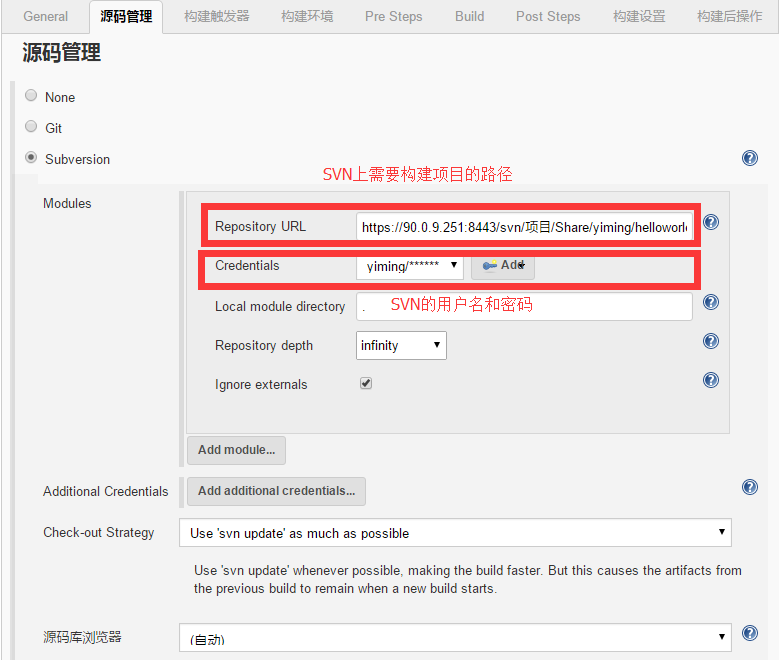




1. General，这一步不必在意，其它选项不用勾选



1. 源码管理



1. 构建触发器



Poll SCM：定时检查源码变更（根据SCM软件的版本号），如果有更新就checkout最新code下来，然后执行构建动作。

Build periodically：定时进行项目构建（它不care源码是否发生变化）。

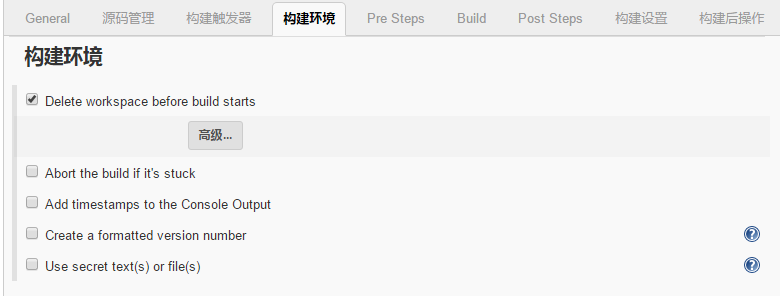
Poll SCM ：\*/10 \* \* \* \*

这是CI 系统中常见的选项。当您选择此选项，您可以指定一个定时作业表达式来定义Jenkins每隔多久检查一下您源代码仓库的变化。如果发现变化，就执行一次构建。例如，表达式中填写\*/10 \* \* \* \*将使Jenkins每隔10分钟就检查一次您源码仓库的变化。表达式中填写0,15,30,45 \* \* \* \*将使Jenkins每隔15分钟就检查一次您源码仓库的变化。

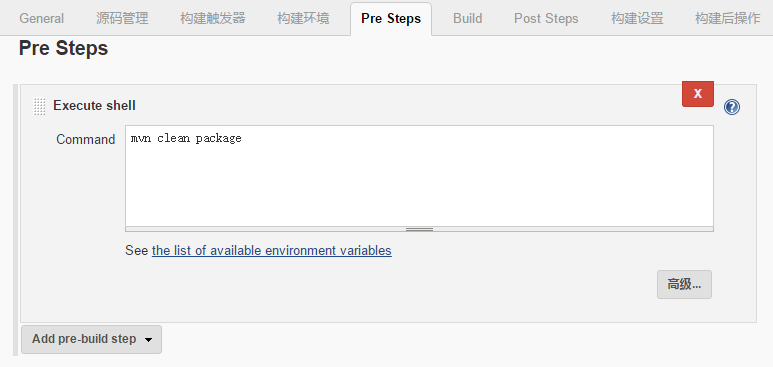
Build periodically：0 \* \* \* \*

第一个参数代表的是分钟 minute，取值 0~59；  
第二个参数代表的是小时 hour，取值 0~23；  
第三个参数代表的是天 day，取值 1~31；  
第四个参数代表的是月 month，取值 1~12；  
最后一个参数代表的是星期 week，取值 0~7，0 和 7 都是表示星期天。  
  
所以 0 \* \* \* \* 表示的就是每个小时的第 0 分钟执行一次构建。

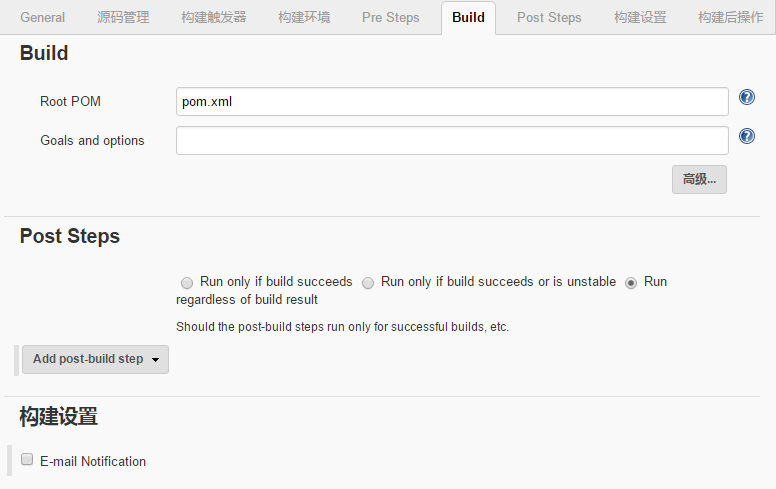
1. 构建环境，构建前清空workspace



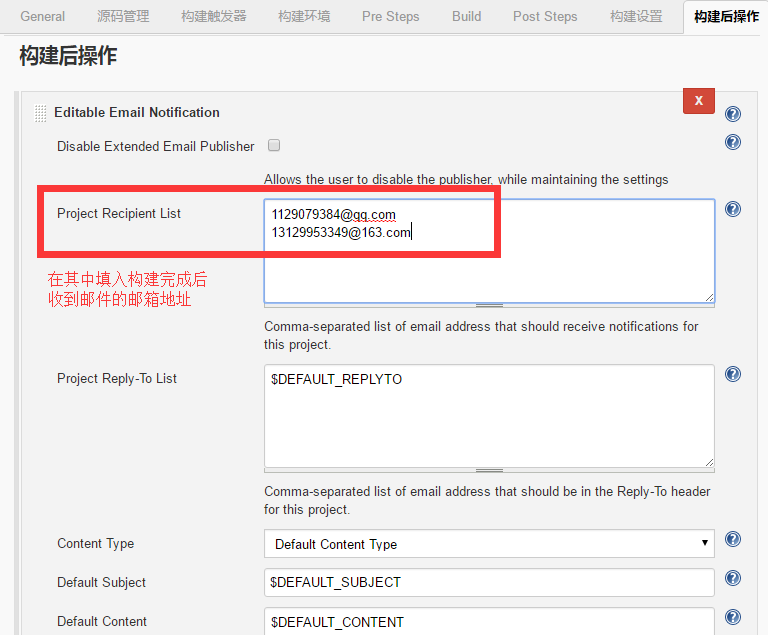
1. Pre Steps，由于是在Linux下，所以选择“Execute shell”，并填入需要对项目进行的操作，如：mvn clean package

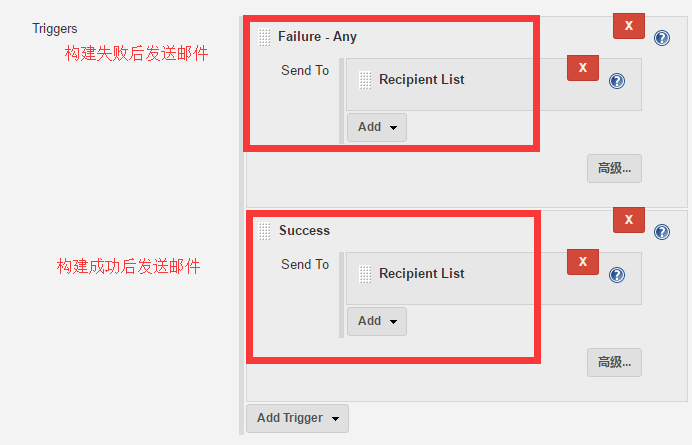


1. 以下三项可不操作

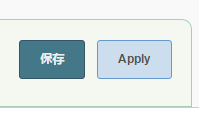


1. 构建后操作，并设置triggers，即在什么情况下发送邮件





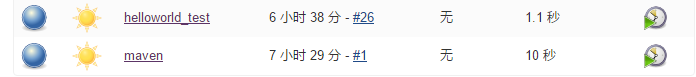
1. 设置完成后，apply并保存



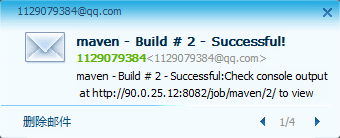
1. 配置保存后，点击“立即构建”



1. 构建成功后，图标为**蓝色**，反之为**红色**



1. 构建后会向你发送邮件



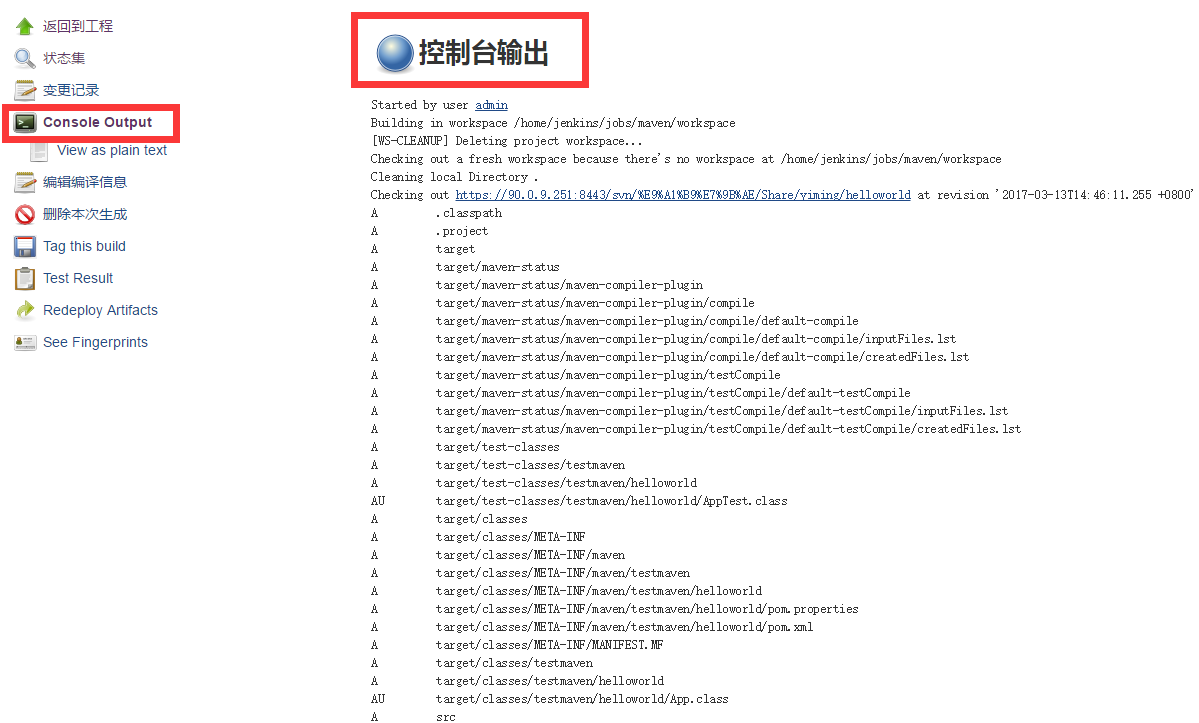
1. 点开项目



1. 点开“Last build”

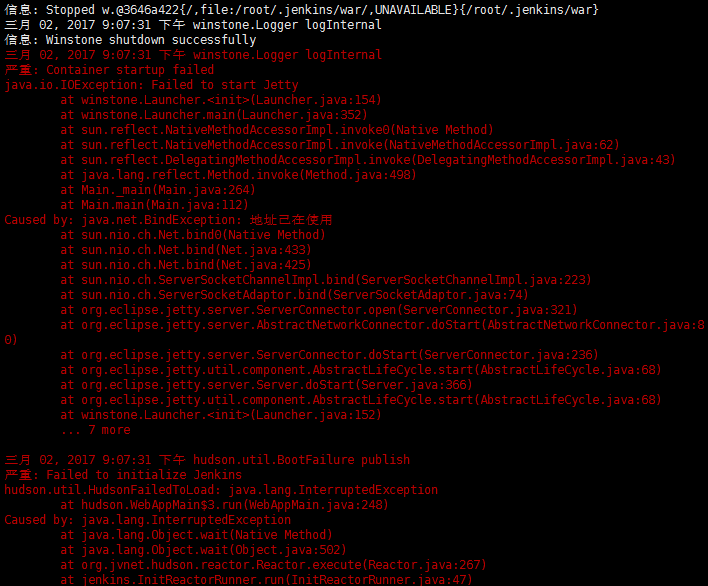


1. 点开“Console Output”，会看到控制台的输出情况



Jenkins注意事项

1. Jenkins启动出现的问题：



 此错误是由于8080端口被占用，

改变端口再次执行，

java -jar jenkins.war --httpPort=$HTTP\_PORT， 例如java -jar jenkins.war --httpPort=8082

启动成功；

后打开[浏览器](http://www.2cto.com/os/liulanqi/)，输入http:/localhost:8082，即可看到jenkins的页面

1. 如果浏览器提示更新密码，一定要选择“否”，否则到时候用户名/密码会更改，统一用admin/123456



1. 给指定用户发送邮件时，指定用户必须开启smtp服务，否则不能接收到邮件

