**持续集成---jenkins**

**修订记录**

**模板版本变更记录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **版本号** | **拟制/修改日期** | **拟制/修改人** | **拟制/修改记录** | **拟制/修改位置** | **批准人** |
| V1.0 | 2018-1-25 | 王磊 | 初次拟制 | 全文 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**工件版本变更记录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **版本号** | **拟制/修改日期** | **拟制/修改人** | **拟制/修改记录** | **拟制/修改位置** | **批准人** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[**1** **前言** 3](#_Toc505689759)

[1.1 编写目的 3](#_Toc505689760)

[1.2 适用范围 3](#_Toc505689761)

[1.3 预期读者 3](#_Toc505689762)

[1.4 参考文档 3](#_Toc505689763)

[**2** **术语和缩略语** 4](#_Toc505689764)

[**3** **Jenkins概述** 4](#_Toc505689765)

[3.1 功能描述 4](#_Toc505689766)

[3.2 安装和使用 4](#_Toc505689767)

[3.3 Jenkins安装与配置 4](#_Toc505689768)

[3.4 JDK安装（略） 4](#_Toc505689769)

[3.5 Tomcat安装（略） 4](#_Toc505689770)

[**4** jenkins相关配置 6](#_Toc505689771)

[4.1 第二章：项目配置及自动化部署 8](#_Toc505689772)

[4.2 发送邮件 13](#_Toc505689773)

[4.3安装CheckStyle插件 16](#_Toc505689774)

[4.4.安装PMD Plug-in插件 17](#_Toc505689775)

[4.5安装FindBugs插件 18](#_Toc505689776)

[4.6项目配置FindBugs 18](#_Toc505689777)

[4.7安装Coverage 插件 19](#_Toc505689778)

**持续集成---jenkins**

# **前言**

## 编写目的

* 整理目前学校端所采用的持续集成方案，梳理汇总文档。
* 便于项目组开发人员和入职新人查阅。

## 适用范围

新学校端项目组

## 预期读者

|  |  |  |
| --- | --- | --- |
| **读者分类** | **阅 读 重 点** | **备注** |
| 项目组 | 全文 |  |
| 入职新人 | 全文 |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 参考文档

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **文档名称** | **日期/版本** | **备注** |
| **1** | 使用Jenkins进行持续集成 |  | https://www.liaoxuefeng.com/article/001463233913442cdb2d1bd1b1b42e3b0b29eb1ba736c5e000 |
| **2** | Analyzing with the SonarQube Scanner |  | https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner+for+Jenkins |
| **3** | DevOps的前世今生 |  | http://www.infoq.com/cn/news/2016/09/learn-devops-from-reports |
| **4** | 以持续集成工具实现DevOps之禅 |  | http://www.infoq.com/cn/news/2016/04/DevOps-continuous-integration-to |

# **术语和缩略语**

持续集成（Continuous Integration，缩写为CI）：是随着敏捷开发一起成长起来的理念，核心是减少缺陷引入，发现和修复之间的时间间隔。做到尽早地规避风险，减少后期代码维护成本和上传的时间人工成本，还有对于版本的控制，保证线上问题的及时回滚。

# **Jenkins概述**

## 功能描述

1.持续、自动地构建/测试软件项目，如CruiseControl与DamageControl。  
2.监控一些定时执行的任务。

## 安装和使用

## **Jenkins安装与配置**

## JDK安装（略）

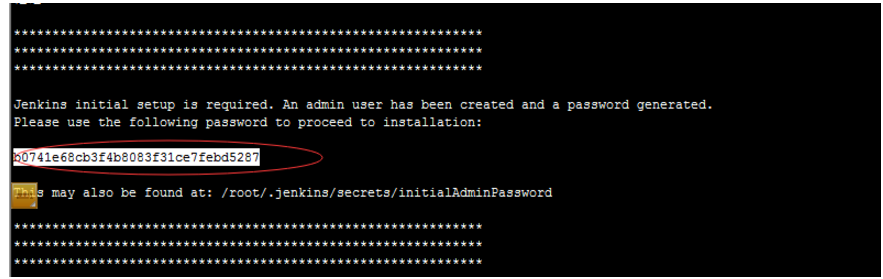
## Tomcat安装（略）

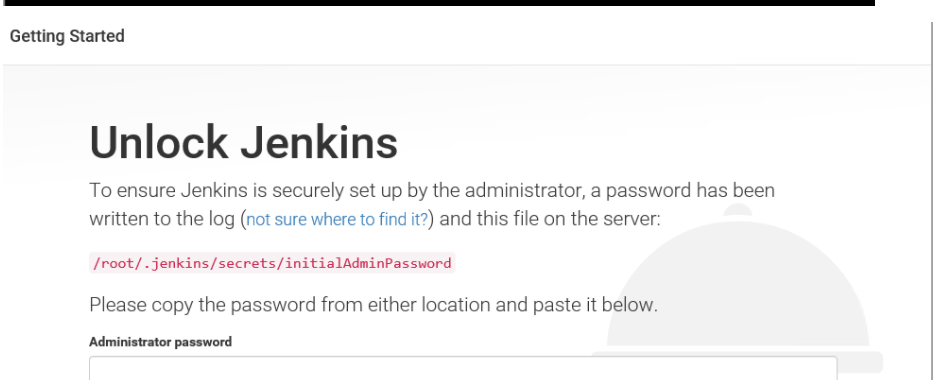
方法1：将jenkins.war包放置webapp目录下，启动tomcat

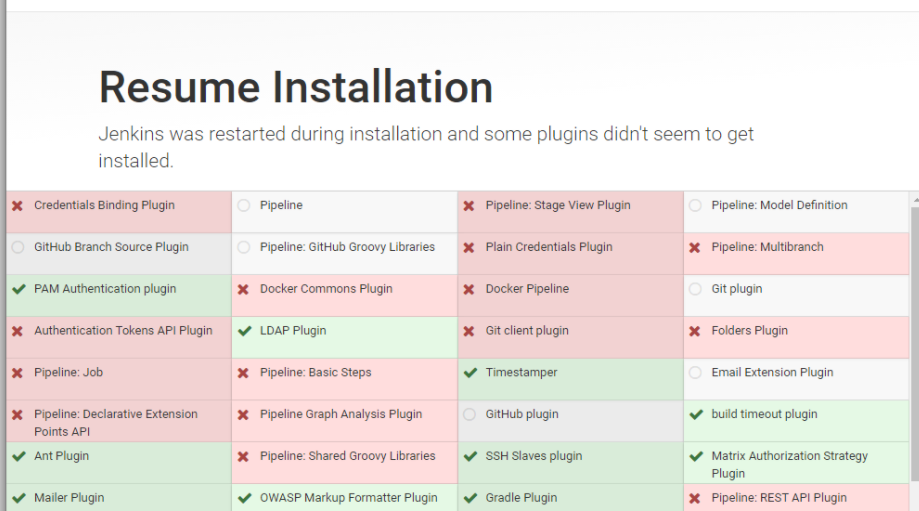
            访问http://ip:port

注：如果报错，则用nohup java -jar jenkins.war --httpPort=8081 &启动进入页面安装插件

输入admin的用户和密码（后台日志有打印）







3.5 .maven安装

tar -xzvf apache-maven-3.3.9-bin.tar.gz

cd %maven%bin

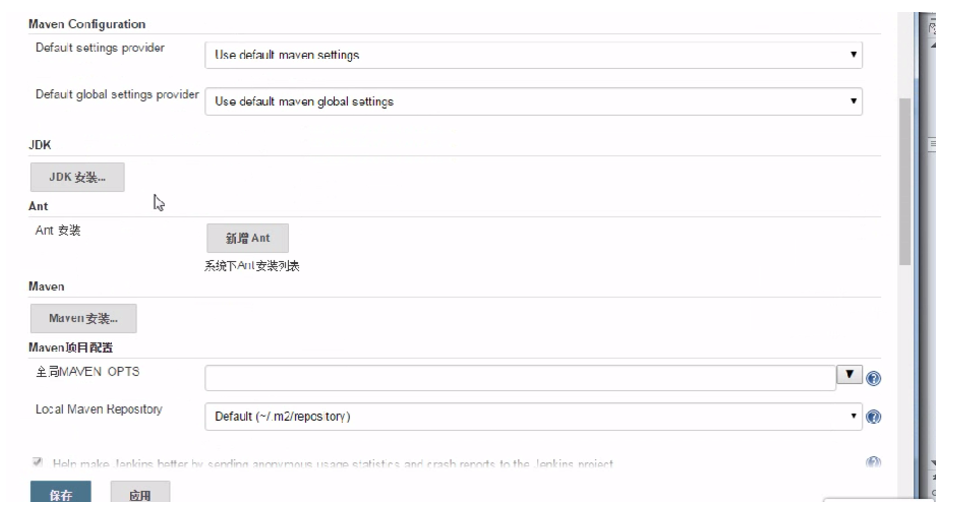
.mvn --help    查看maven的用法

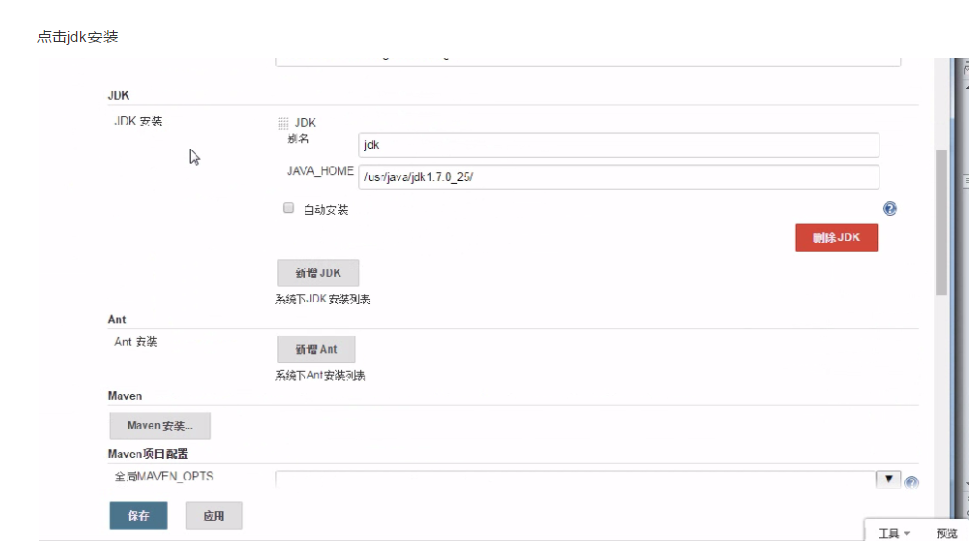
# jenkins相关配置

依次点击”系统管理”–》”系统设置”









点击jdk安装

不勾选自动安装，JAVA\_HOME配置JDK实际安装的目录

点击Maven安装

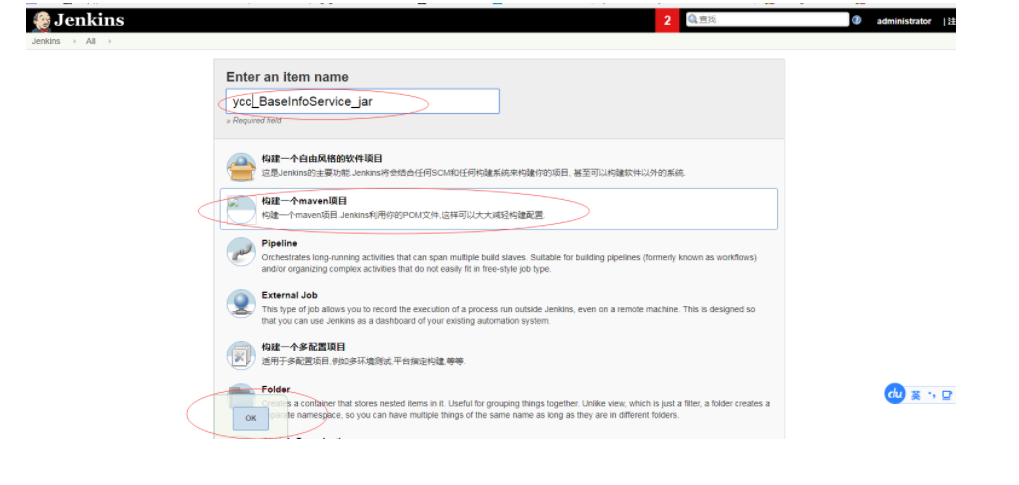
不勾选自动安装，MAVEN\_HOME配置maven实际安装的目录

## **第二章：项目配置及自动化部署**

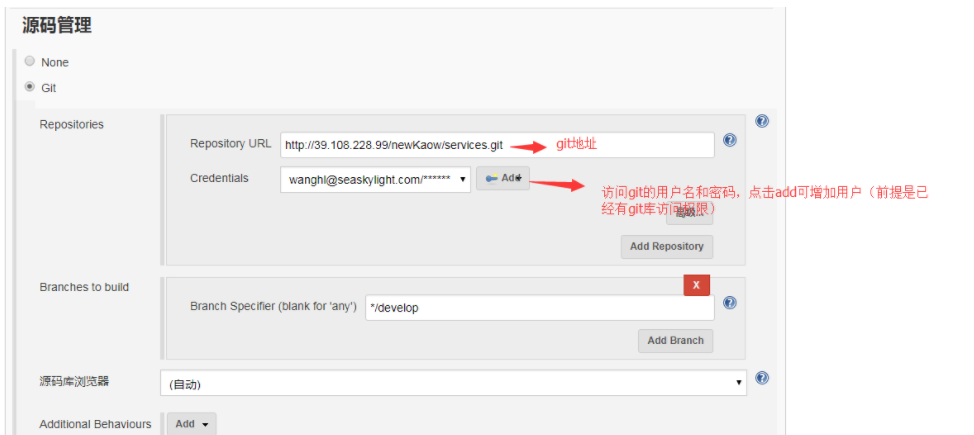
注：Maven项目相关插件安装：（请前往“系统管理”--》管理插件下载且安装）

[Maven Dependency Update Trigger](http://wiki.jenkins-ci.org/display/JENKINS/Maven+Dependency+Update+trigger)    [Maven Info Plugin](https://wiki.jenkins-ci.org/display/JENKINS/Maven+Info+Plugin)     [Maven Repository Server Plugin](http://wiki.jenkins-ci.org/display/JENKINS/Jenkins+Maven+Repository+Server)

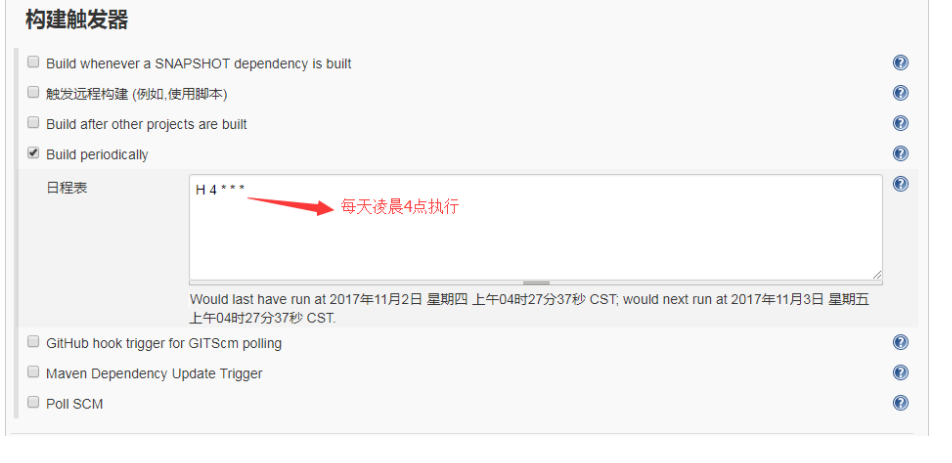
1. 点击开始创建一个新任务，配置Item名称，选择构建一个maven项目，点击OK



2.源码管理配置



3.构建触发器配置



注：schedule的配置规则是由5个空格隔开的字符组成，从左到右分别代表：分、时、天、月、年。\*号代表所有

扩展：如果该项目依赖另外一个项目自动构建，则按如下配置（例如，该项目依赖schtest\_commonModel\_jar项目构建成功后自动构建）



4.构建环境配置

配置在build之后执行一个自动化部署脚本，在这个之前需要安装[SSH plugin](http://wiki.jenkins-ci.org/display/JENKINS/SSH+plugin)插件，并且配置SSH remote hosts

4.1 安装[SSH plugin](http://wiki.jenkins-ci.org/display/JENKINS/SSH+plugin)：系统管理--》管理插件--》可选插件--》搜索[SSH plugin](http://wiki.jenkins-ci.org/display/JENKINS/SSH+plugin)插件，安装

4.2 配置SSH remote hosts：

系统管理--》系统设置--》配置SSH remote hosts

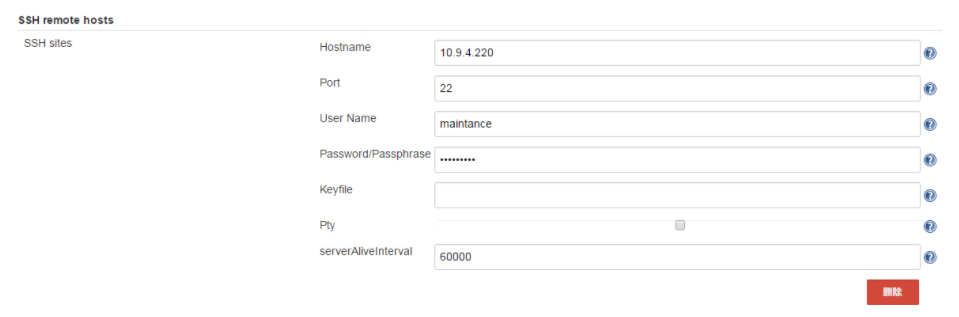
Hostname：输入IP或hostname

Port:连接端口

User Name：用户名

Password/Passphrase：用户密码

serverAliveInterval：服务存活间隔



4.3 项目构建环境配置



shell脚本如下：  $ cat   BaseInfoService\_auto\_crontab.sh

#!/bin/bash

#设定变量

IP=10.9.4.231

IP2=10.9.4.219

USER=schtest

PASSWD='\*\*\*\*\*\*\*\*'

SERVICE=BaseInfoService

T\_WAR=.jar

SFILE\_TEST\_DIR=BaseInfoService-1.0.0-SNAPSHOT

DDIR=/opt/schtest/school\_hedu/BaseInfoService

SDIR=/opt/kaow/.jenkins/jobs/schtest\_BaseInfoService\_jar/workspace/BaseInfoService/target

#第1套实例ssh登录

sshpass -p ${PASSWD} ssh -t -t ${USER}@${IP} -p 22 -o StrictHostKeyChecking=no <<EOF

sleep 5

cd $DDIR

sh kill\_${SERVICE}.sh

if [ $? -eq 0 ];then

sleep 5

mv ${SFILE\_TEST\_DIR}${T\_WAR} ${SFILE\_TEST\_DIR}${T\_WAR}\_$(date +%Y%m%d%H%M%S)

exit

fi

EOF

#第2套实例ssh登录

sshpass -p ${PASSWD} ssh -t -t ${USER}@${IP2} -p 22 -o StrictHostKeyChecking=no <<EOF

sleep 5

cd $DDIR

sh kill\_${SERVICE}.sh

if [ $? -eq 0 ];then

sleep 5

mv ${SFILE\_TEST\_DIR}${T\_WAR} ${SFILE\_TEST\_DIR}${T\_WAR}\_$(date +%Y%m%d%H%M%S)

exit

fi

EOF

#第1套实例scp jar文件

sshpass -p ${PASSWD} scp ${SDIR}/${SFILE\_TEST\_DIR}${T\_WAR} ${USER}@${IP}:${DDIR}

sleep 5

#第2套实例scp jar文件

sshpass -p ${PASSWD} scp ${SDIR}/${SFILE\_TEST\_DIR}${T\_WAR} ${USER}@${IP2}:${DDIR}

sleep 5

###########

#第1套实例ssh登录

sshpass -p ${PASSWD} ssh -t -t ${USER}@${IP} -p 22 -o StrictHostKeyChecking=no <<EOF

sleep 5

cd $DDIR

sh start.sh

if [ $? -eq 0 ];then

sleep 5

exit

fi

EOF

#第2套实例ssh登录

sshpass -p ${PASSWD} ssh -t -t ${USER}@${IP2} -p 22 -o StrictHostKeyChecking=no <<EOF

sleep 5

cd $DDIR

sh start.sh

if [ $? -eq 0 ];then

sleep 5

exit

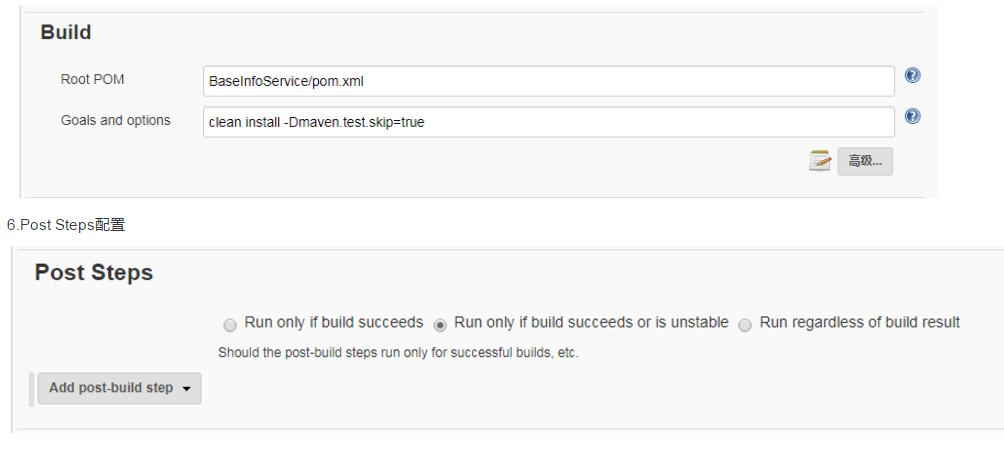
fi

EOF

5.Pre Steps 配置

Root POM：配置项目pom.xml路径

Goals and options：配置maven执行命令



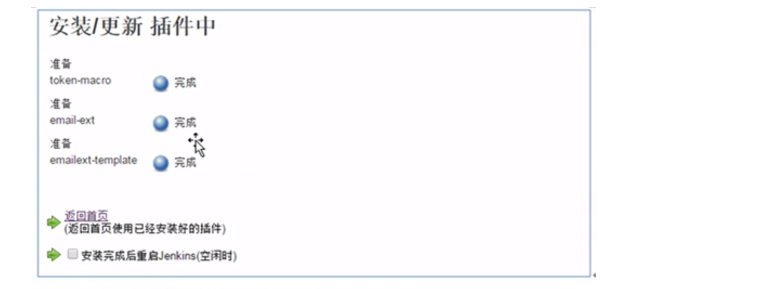
6.Post Steps配置

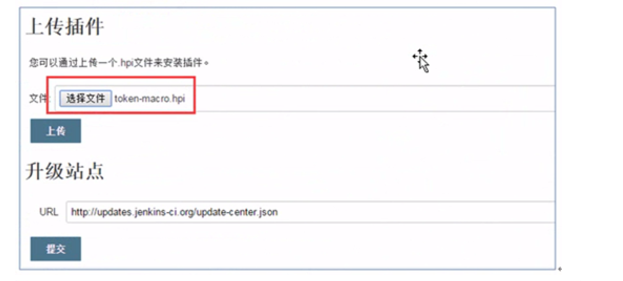
## 发送邮件

1.安装插件[Email Extension Plugin](http://wiki.jenkins-ci.org/display/JENKINS/Email-ext+plugin)

方法1：直接去“系统管理”--》“管理插件”--》可选插件搜索[Email Extension Plugin](http://wiki.jenkins-ci.org/display/JENKINS/Email-ext+plugin)安装即可

方法2：在<https://wiki.jenkins-ci.org/display/JENKINS> 官网下载[Email Extension Plugin](http://wiki.jenkins-ci.org/display/JENKINS/Email-ext+plugin)插件离线安装



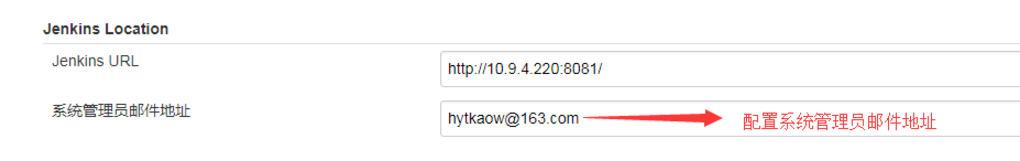




2.系统配置邮件

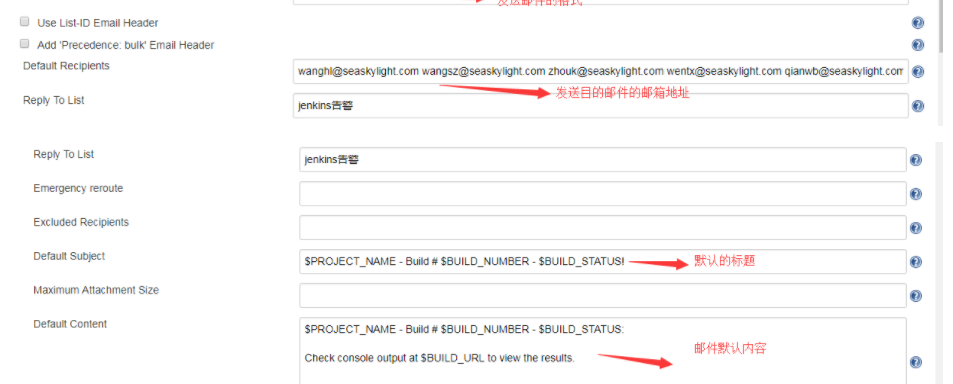
点击“系统管理”–》“系统设置”

2.1配置系统管理员邮件地址



2.2找到Extended E-mail Notification区域

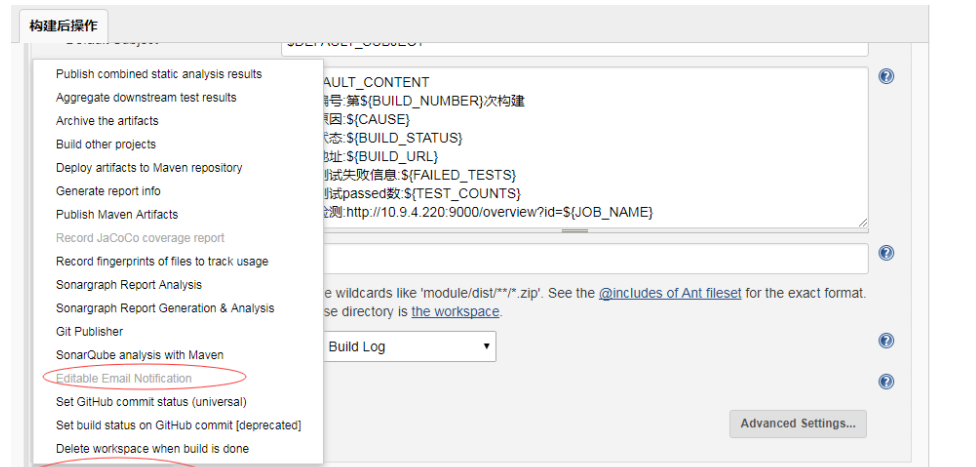




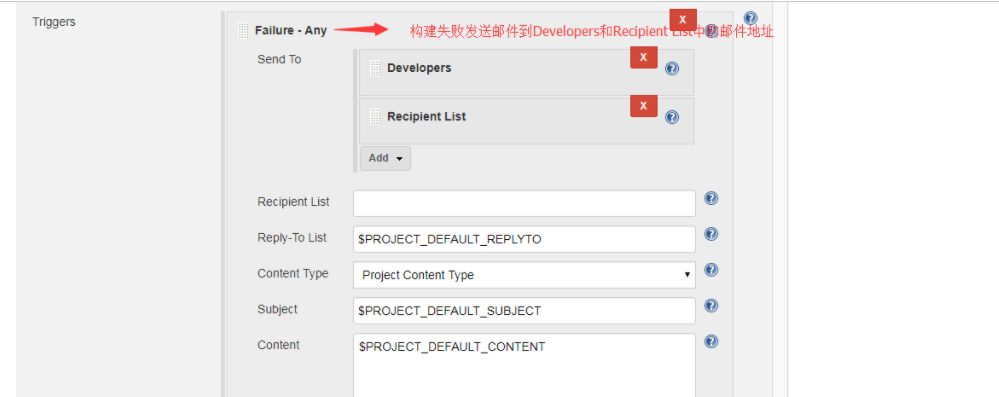
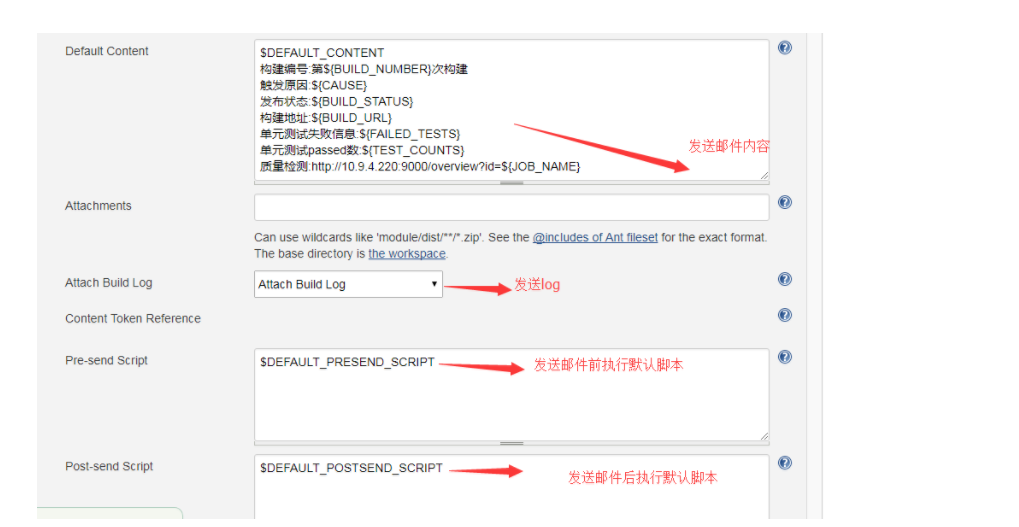


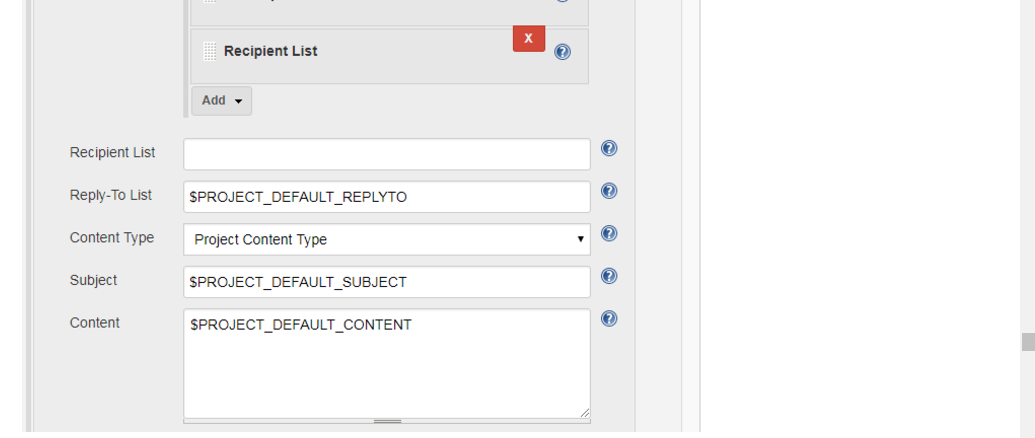
2.3 项目邮件相关配置

在项目配置页面，找到构建后操作区域，点击“增加构建后操作步骤”，选择“Editable Email Notification”







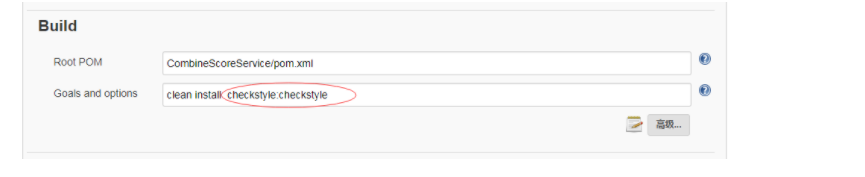


### 4.3安装CheckStyle插件

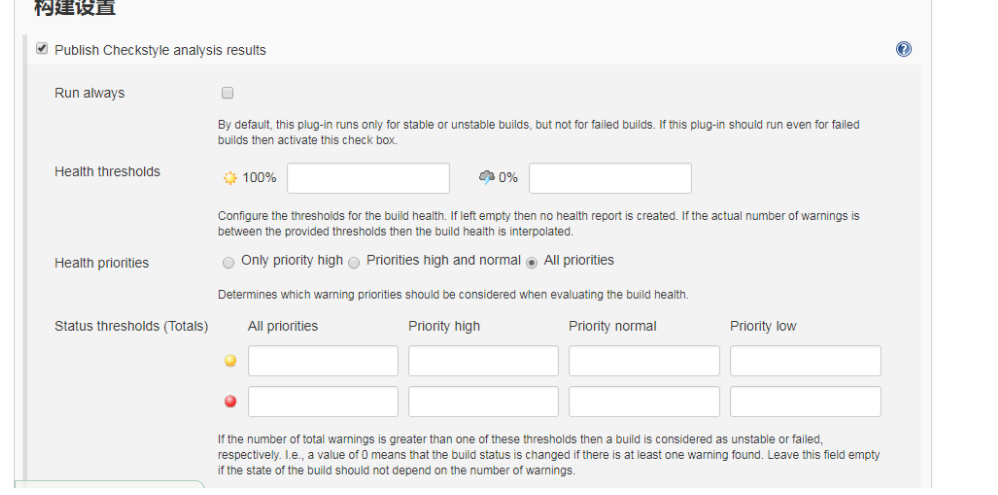
依次点击"系统管理"--》“管理插件”--》可选插件，搜索[Checkstyle Plug-in](http://wiki.jenkins-ci.org/x/GYCGAQ)，点击“直接安装”

4.1.2.项目配置CheckStyle

在项目配置Build区域的Goals and options字段配置checkstyle:checkstyle即可



在构建设置区域，勾选Publish Checkstyle analysis results单选框，需要填写规则，在里面配置即可

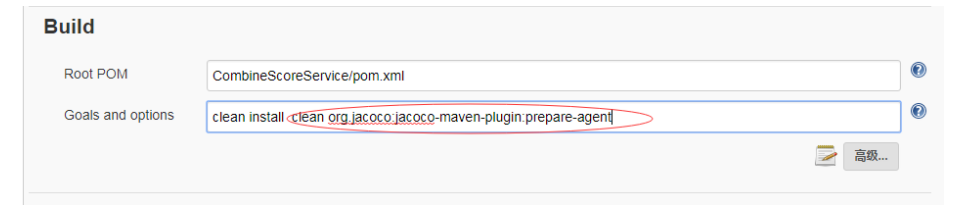


### 4.4.安装[PMD Plug-in](http://wiki.jenkins-ci.org/x/GAAHAQ)插件

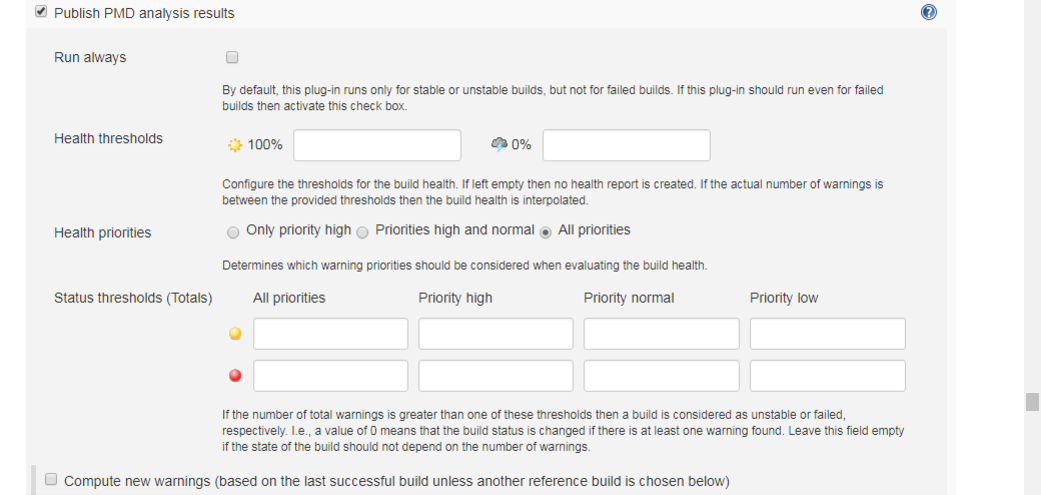
依次点击"系统管理"--》“管理插件”--》可选插件，搜索[PMD Plug-in](http://wiki.jenkins-ci.org/x/GAAHAQ)，点击“直接安装”

#### 4.2.2.项目配置PMD

在项目配置Build区域的Goals and options字段配置 pmd:pmd即可



在构建设置区域，勾选Publish PMD analysis results单选框，需要填写规则，在里面配置即可

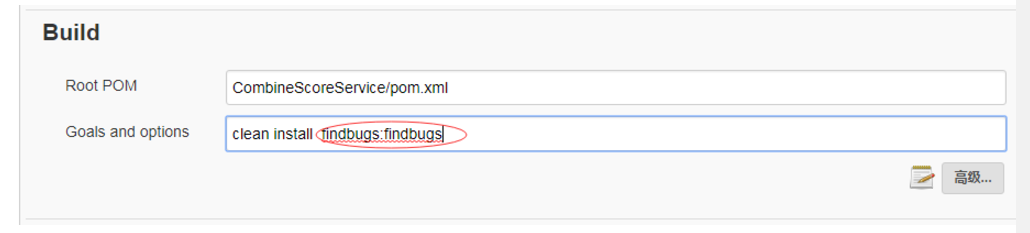


### 4.5.安装[FindBugs](http://10.9.4.220:8081/view/school_devday_jar/job/schoolday_CombineScoreService_jar/53/findbugsResult)插件

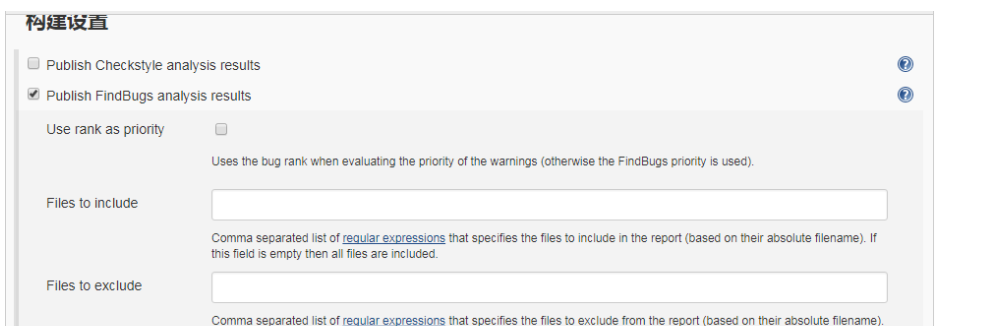
依次点击"系统管理"--》“管理插件”--》可选插件，搜索[FindBugs Plug-in](http://wiki.jenkins-ci.org/x/GYAs)，点击“直接安装”

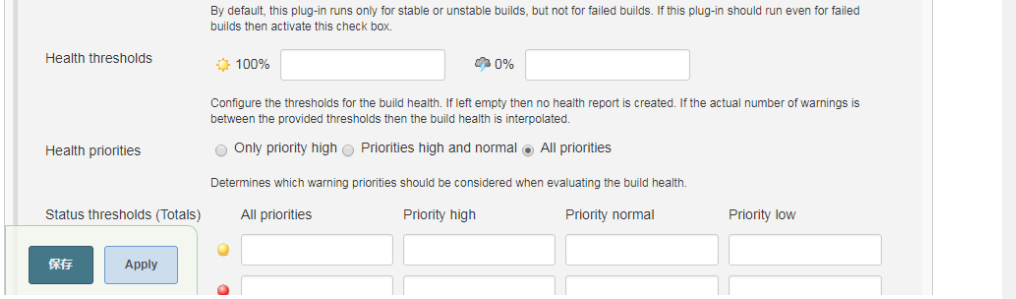
### 4.6.项目配置[FindBugs](http://10.9.4.220:8081/view/school_devday_jar/job/schoolday_CombineScoreService_jar/53/findbugsResult)

在项目配置Build区域的Goals and options字段配置 findbugs:findbugs即可



在构建设置区域，勾选Publish FindBugs analysis results单选框，需要填写规则，在里面配置即可



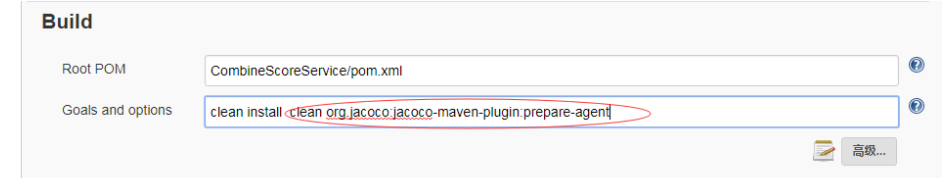


### 4.7.安装[Coverage](http://10.9.4.220:8081/view/school_devday_jar/job/schoolday_CombineScoreService_jar/53/jacoco)插件

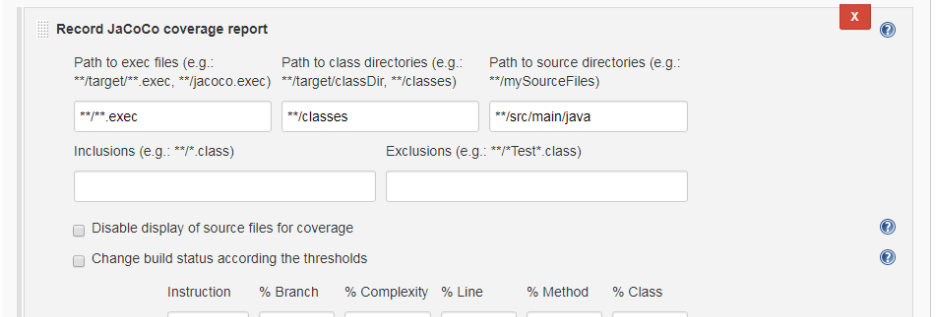
依次点击"系统管理"--》“管理插件”--》可选插件，搜索JaCoCo plugin，点击“直接安装”

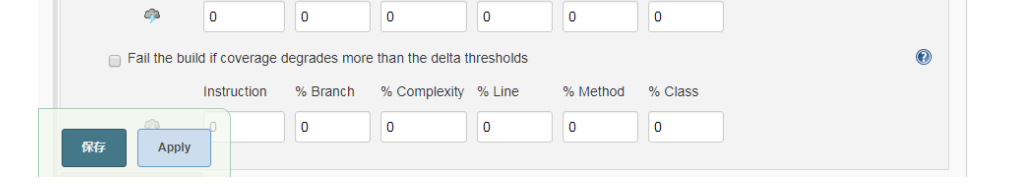
4.4.2.项目配置[Coverage](http://10.9.4.220:8081/view/school_devday_jar/job/schoolday_CombineScoreService_jar/53/jacoco)

在项目配置Build区域的Goals and options字段配置clean org.jacoco:jacoco-maven-plugin:prepare-agent即可



在构建后操作区域，点击“增加构建后操作”，选择"Record JaCoCo coverage report"，需要填写规则，在里面配置即可





**---------------- < 完 > ----------------**