**Jenkins日常操作手册**

张义明

2015-05-22

目录

[修订历史 3](#_Toc420584946)

[前言 4](#_Toc420584947)

[1、日常操作 4](#_Toc420584948)

[1.1 登录 4](#_Toc420584949)

[1.2 打包发布 5](#_Toc420584950)

[1.3 构建完毕 7](#_Toc420584951)

[1.4 DCMD后续处理 8](#_Toc420584952)

[2、添加CI流程 8](#_Toc420584953)

[2.1 Jenkins流程样例 8](#_Toc420584954)

[2.1.1 template\_byHand 8](#_Toc420584955)

[2.1.2 template\_auto 8](#_Toc420584956)

[2.2 添加自动代码检查流程 9](#_Toc420584957)

[a) 新建 9](#_Toc420584958)

[b) 选择“复制已有的Item” 10](#_Toc420584959)

[c) 修改配置 10](#_Toc420584960)

[d) 保存 11](#_Toc420584961)

[2.3 添加手动打包发布流程 15](#_Toc420584962)

[a) 新建 15](#_Toc420584963)

[b) 选择“复制已有的Item” 15](#_Toc420584964)

[c) 修改配置 16](#_Toc420584965)

[d) 保存 17](#_Toc420584966)

# 修订历史

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 修订日期 | 修订人 | 修订说明 |
| V0.1 | 2015-05-22 | 张义明 | 创建文档 |
| V0.2 | 2015-05-25 | 张义明 | 完善样例使用说明部分内容 |
| V0.3 | 2015-05-26 | 张义明 | 完善内容 |
| V0.4 | 2015-05-28 | 张义明 | 完善模板项目，修改说明 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

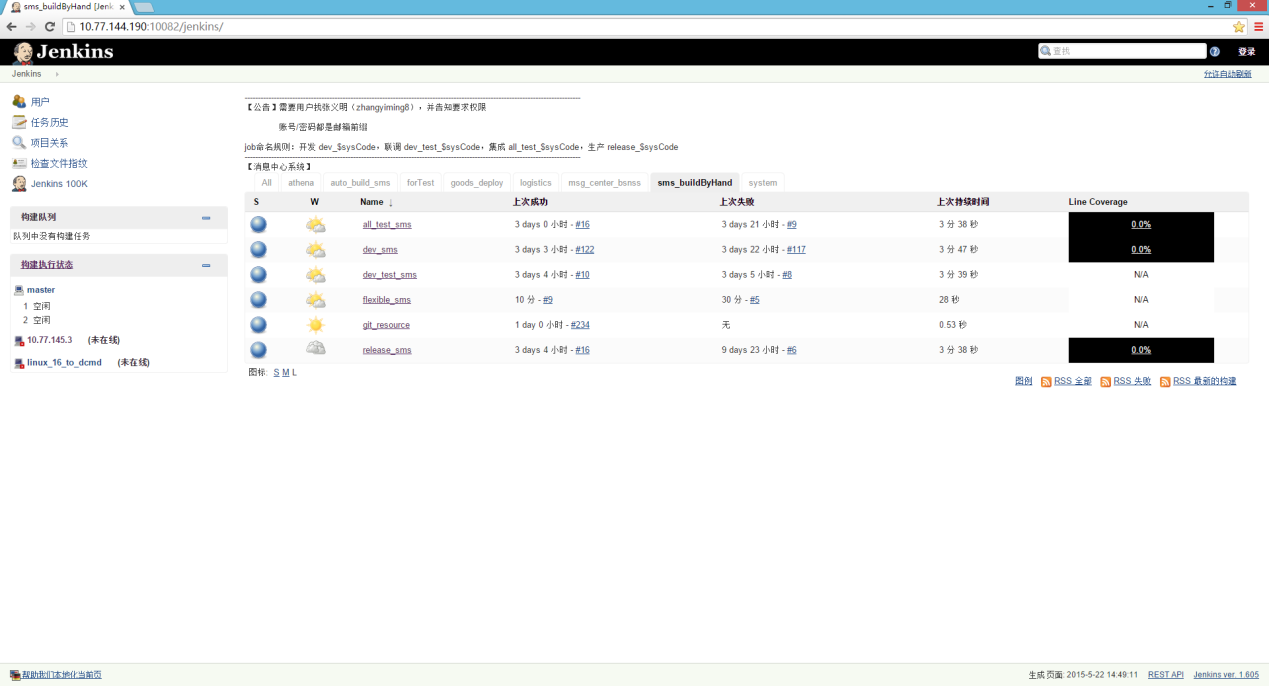
# 前言

此文档主要目的是为了日常构建操作、添加CI流程时使用的指导手册，如果想详细的了解Jenkins使用与配置，参见《Jenkins安装与使用.docx》。

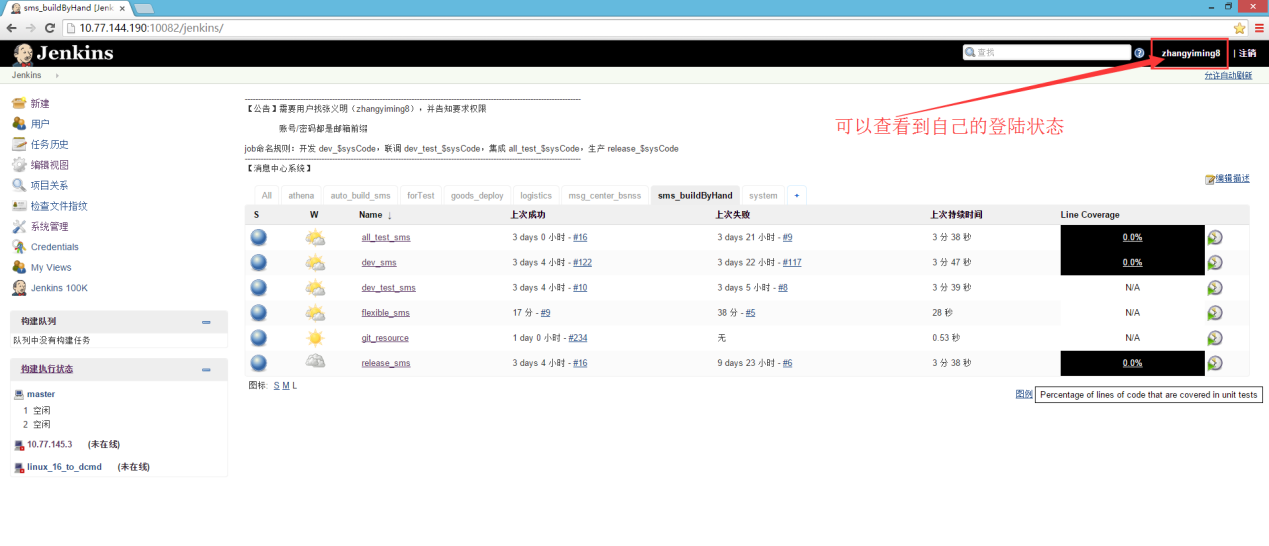
# 1、日常操作

## 1.1 登录

访问地址<http://10.77.144.190:10082/jenkins/>

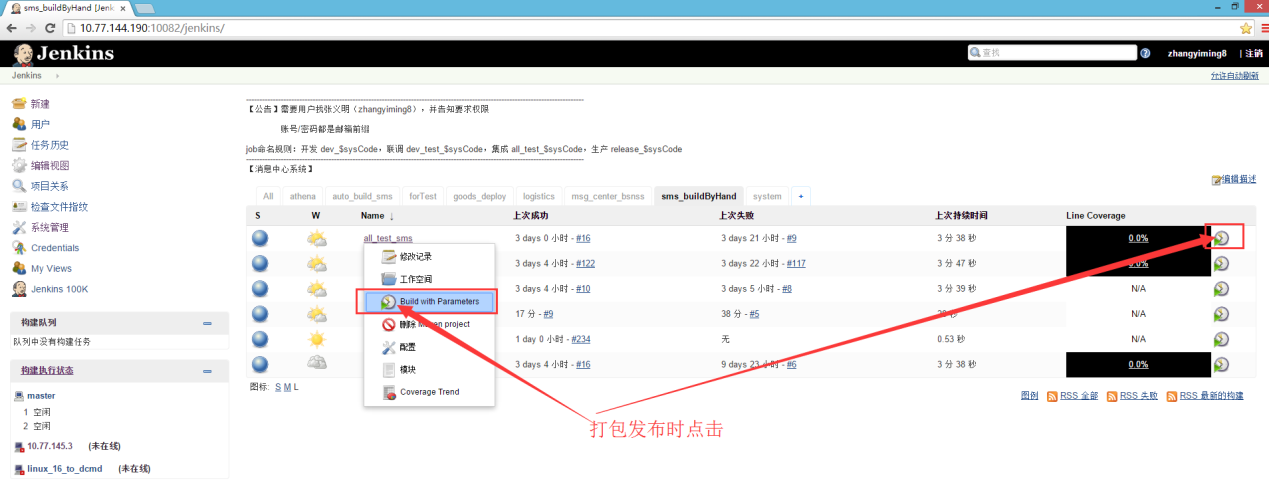


点击右上角的“登录”，输入账号进行登录，账号和密码都是CTX，如果登陆不了或者没有账户，联系张义明（[zhangyiming8@wanda.cn](mailto:zhangyiming8@wanda.cn)），登陆之后才能进行项目的构建操作，未登录用户没有构建权限。

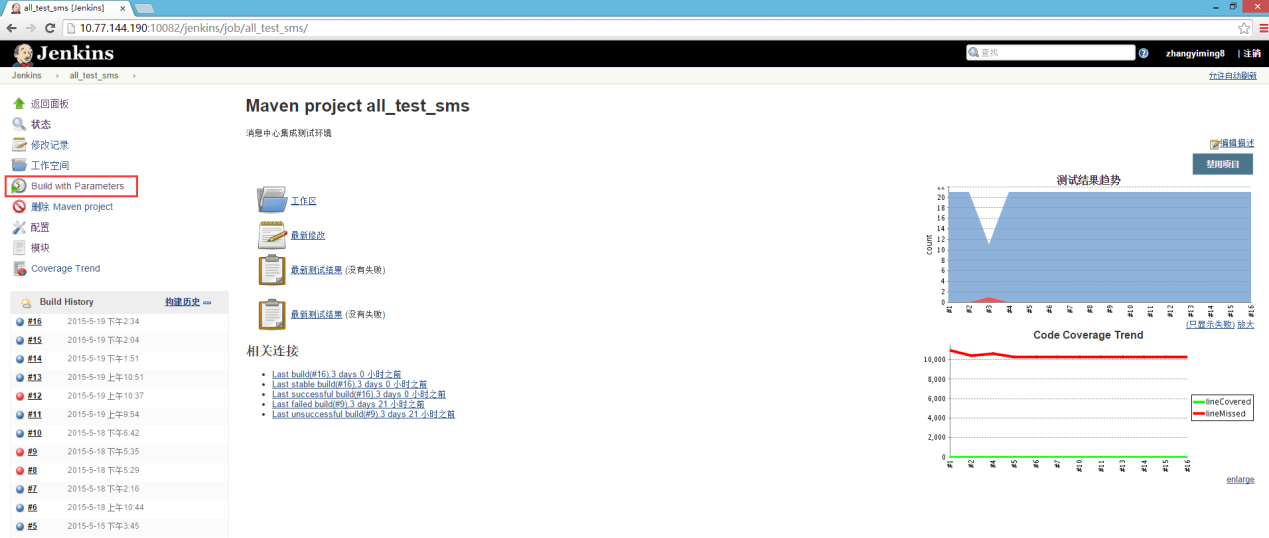


### 1.2 打包发布

已经登录的用户具有打包发布权限。登录后，可以点击项目后面的按钮进行打包操作

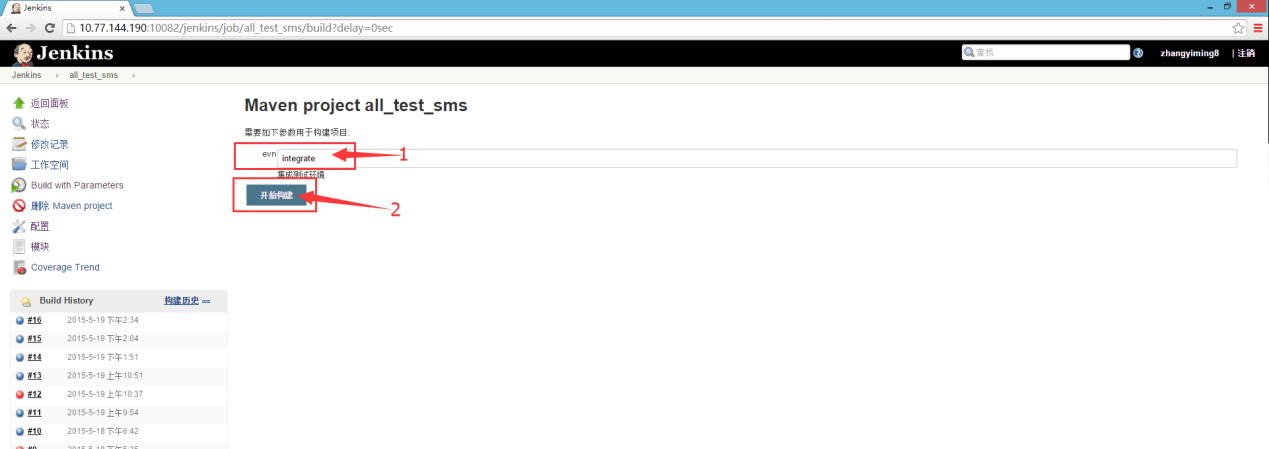


或者在项目界面点击：



如果登录后没有此按钮，找Jenkins管理员申请权限。

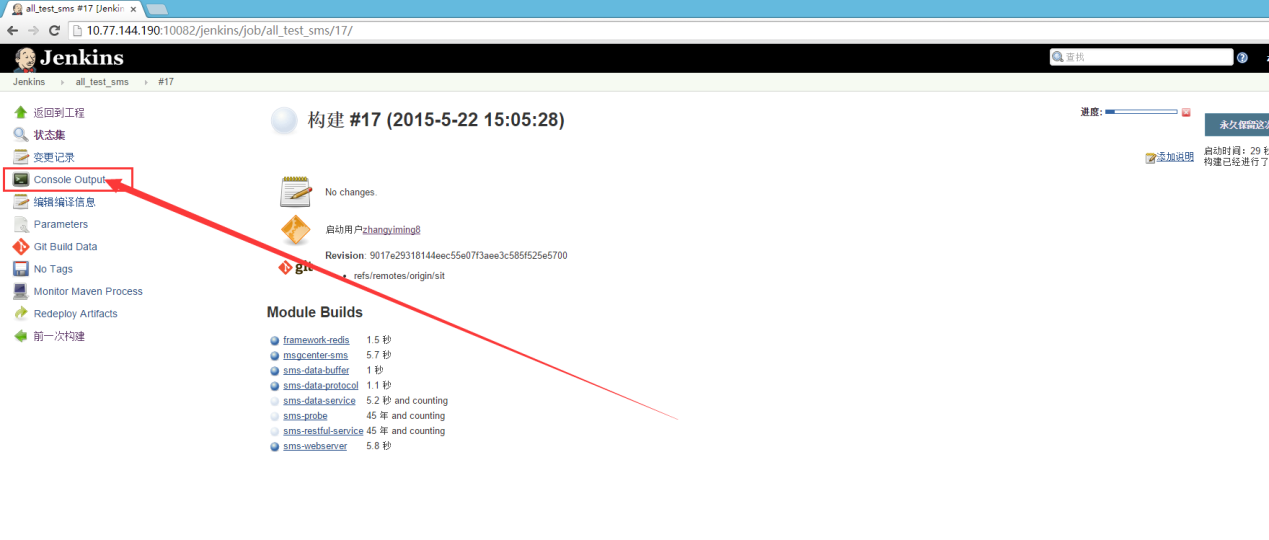
点击构建后，会进入指定参数界面



再点击“开始构建”即可，在左侧可以看到项目构建进度，此例中指定的是构建哪个环境的发布包（开发环境、集成环境、生产环境、联调环境，具体值要看项目组自己的定义）

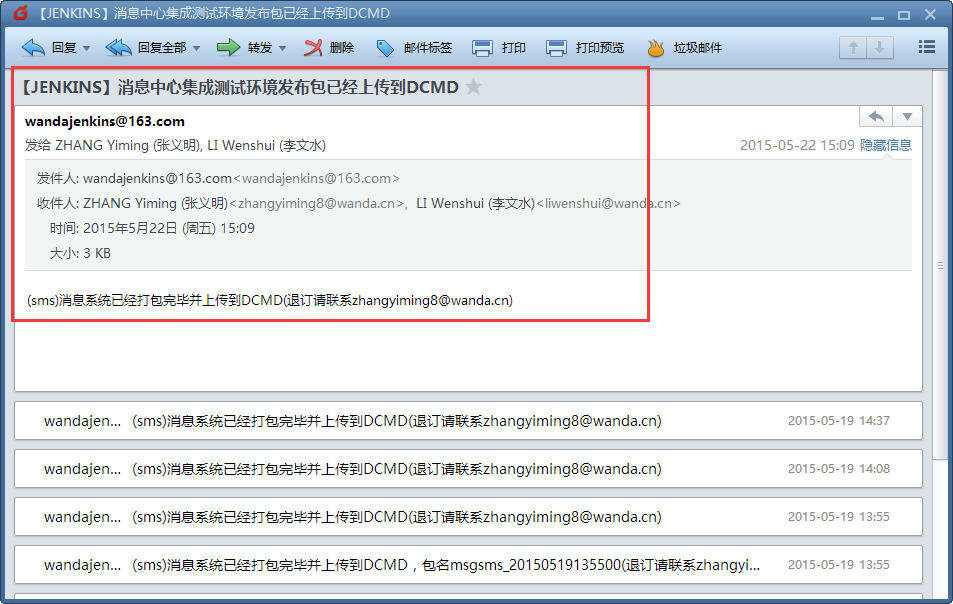


也可以点击查看项目构建时的日志输出

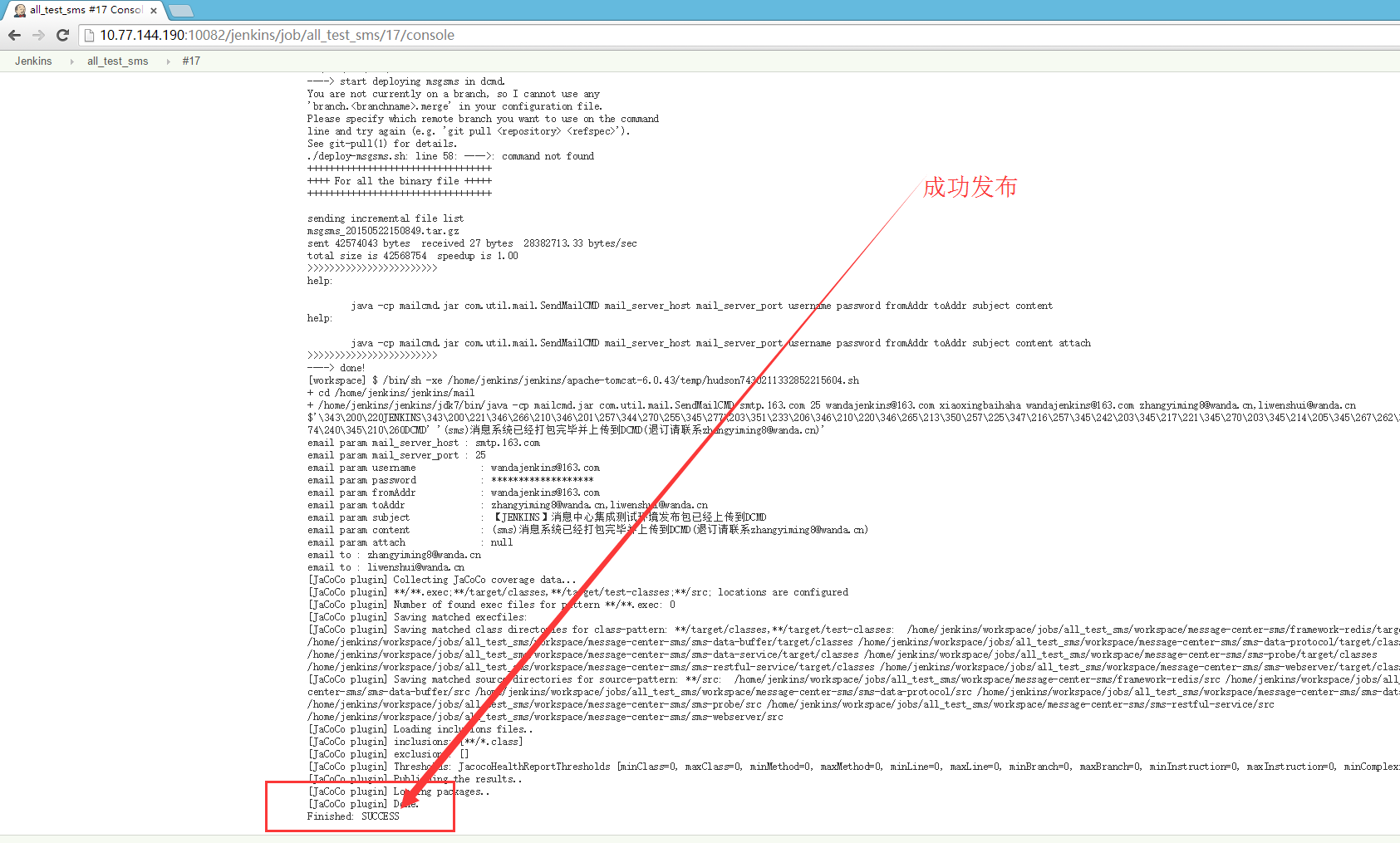


### 1.3 构建完毕

收到邮件提醒后，就可以知道打包发布完成



可以查看日志中是否正常。



如果失败了，查看日志分析具体原因，解决后，再次执行打包发布操作，直至成功。

### 1.4 DCMD后续处理

收到邮件提醒后，如果打包发布成功，可以登陆DCMD尽心后续处理。

# 2、添加CI流程

## 2.1 Jenkins流程样例

已预置了多种CI流程样例（在template页签）：

### 2.1.1 template\_hand\_war

* 1. 作为手动构建打包并发布工程的引用模板；(针对打War包)
  2. 由于手动构建时需要发布到dcmd，所以此模板包含代码编译、代码检查、打包、发布到dcmd、邮件提醒（附件是代码检查的报告）；

### 2.1.2 template\_hand\_jar

* 1. 作为手动构建打包并发布工程的引用模板；(针对打Jar包)
  2. 由于手动构建时需要发布到dcmd，所以此模板包含代码编译、代码检查、打包、发布到dcmd、邮件提醒（附件是代码检查的报告）；

### 2.1.3 template\_auto

* 1. 作为自动化构建及测试工程引用模板；
  2. 自动构建工程需要包含代码编译、跑单元测试用例、邮件提醒（如果编译出错或者单元测试用例出错会发邮件提醒）；

## 2.2 添加自动代码检查流程

新建时参照template\_auto工程添加。具体操作方法：(以交易定时任务为例)

### a) 新建

登陆用户可以点击左侧的“新建”进行增加CI流程操作(如果没有该功能，请联系Jenkins管理员申请)。

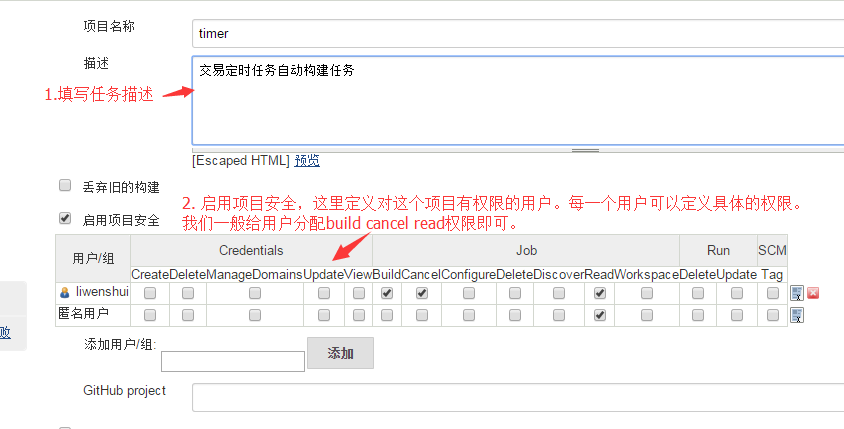


Item名称必须跟gitlab中的项目名称一致，否则无法自动触发。

### b) 修改配置

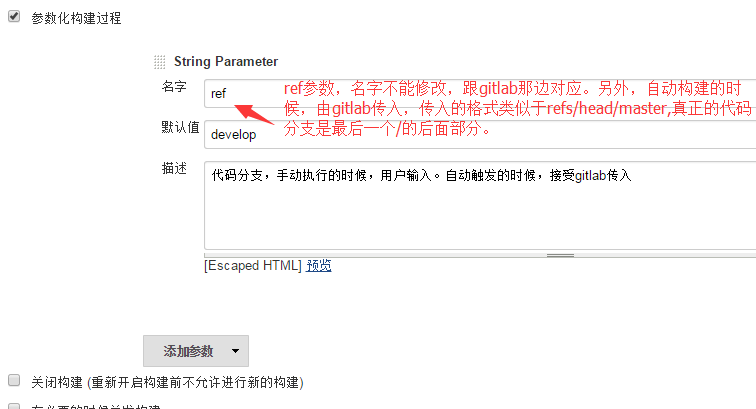
点击OK之后，进入修改配置界面：

1. 任务构建策略，权限配置



（默认加一个匿名用户，分配读权限。不要问为什么，照做就行）

2.参数化配置



3. 配置身份令牌和动态根据分支下载代码



1. 配置pom文件和发送邮件



### c) 保存

配置完毕后点击“保存”即可开始构建(可以先手动构建看脚本配置是否正确)

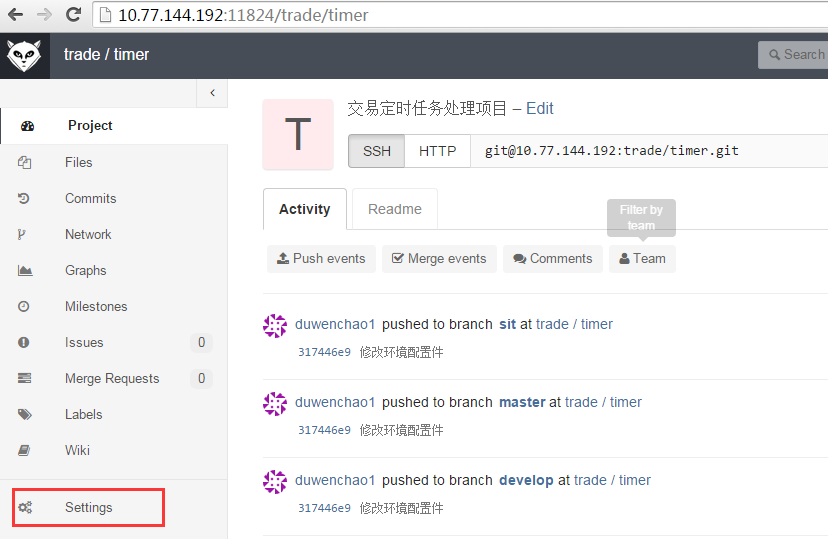
后续动作：为了支持提交代码后，流程即开始自动构建，需要项目配置时在“触发远程构建”处打钩，支持远程调用（这个上面已经做过了，还需要做一个事情）



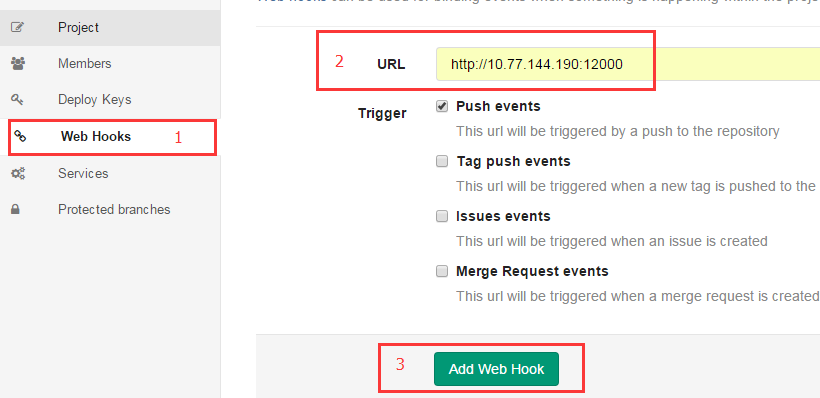
另外一个事情是（方案1，推荐方案）

在gitlab那边加上触发该任务启动的条件：

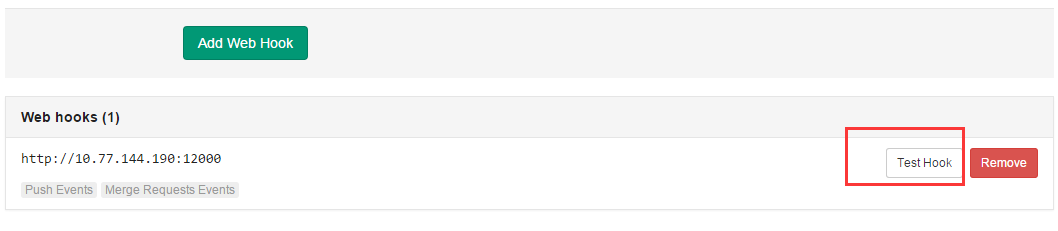
1. 进入gitlab，找到该任务对应的项目，我就找到交易定时任务项目，如下图：



1. 点击Settings, 然后点击 Web Hooks，在URL里面输入：<http://10.77.144.190:12000>最后点击Add Web Hook按钮。如下图：



1. 点击 Test Hook，看看能不能自动触发我们刚新建的任务启动。



（方案2）

添加如下hooks脚本（目的：支持开发人员提交代码到不同分支，自动构建检查对应分支代码）

curl -d "branchName=${1:11}&token=goodscentercommit" [http://10.77.144.190:10082/jenkins/view/goods\_deploy/job/auto\_build\_goodscenterv2/buildWithParameters](http://10.77.144.190:10082/jenkins/view/auto_build_sms/job/auto_checkcode_byParams/buildWithParameters)

这个脚本需要修改四个地方（红色标出）：

branchName ：跟jenkins里面配置的参数名保持一致（见b)修改配置第2项参数化配置）

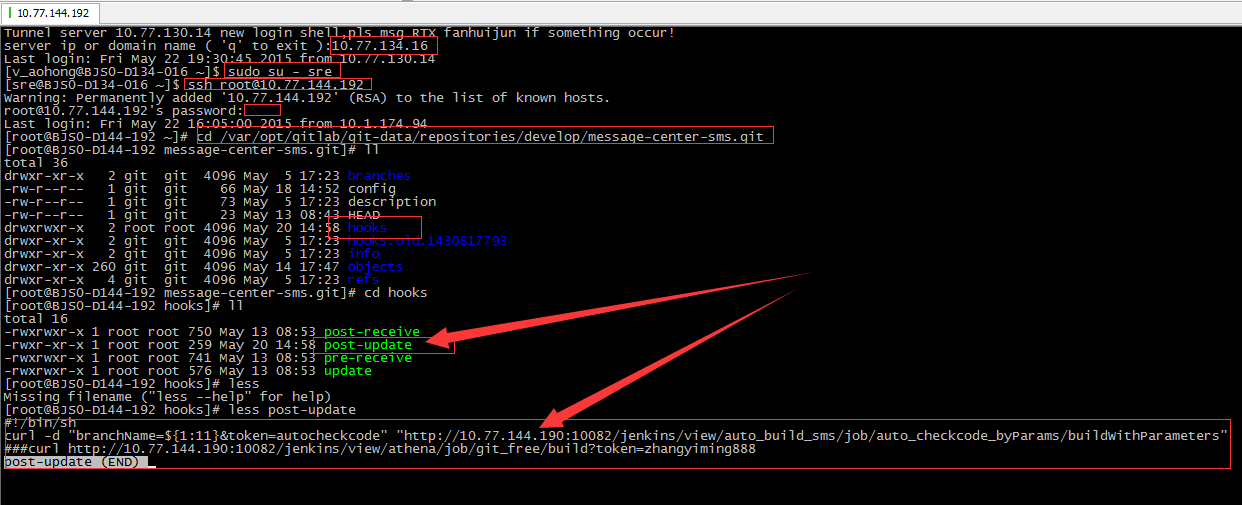
表示代码分支。

goodscentercommit：跟jenkins里面的身份令牌一致（见b)修改配置第3项配置身份令牌）

goods\_deploys ： Jenkins项目所在的页签名称

auto\_build\_goodscenterv2 ： Jenkins项目名称

添加hooks脚本如下图所示：

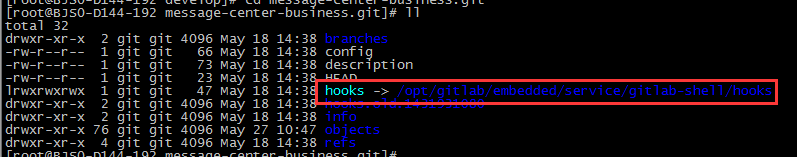


现在可以提交代码到git，查看下Jenkins是不是自动开启了任务呢？

【**修改hooks时注意**】

在gitlab上添加hooks脚本时一定要注意

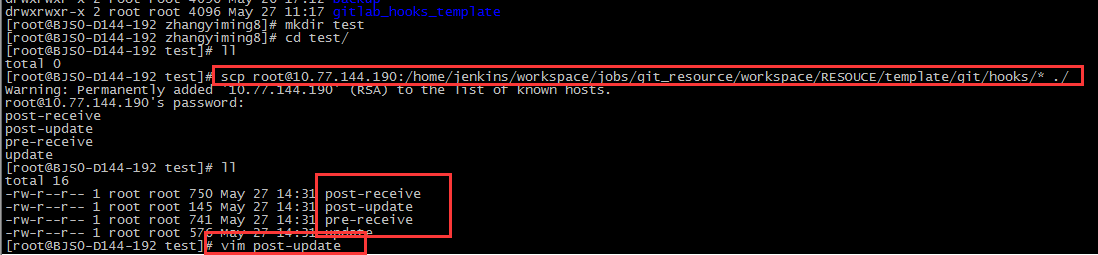
* + **在项目的gitlab上找到该项目对应的git仓储下的hooks目录，如果该目录是个快捷方式**



**一定不要**直接cd hooks修改该目录下的文件脚本，千万注意。

**正确的做法是**删除该快捷方式，手动创建hooks目录，[然后从root@10.77.144.190:/home/jenkins/workspace/jobs/git\_resource/workspace/RESOUCE/template/git/hooks](mailto:然后从root@10.77.144.190:/home/jenkins/workspace/jobs/git_resource/workspace/RESOUCE/template/git/hooks)拷贝hooks下的所有文件到此项目的hooks目录（刚刚手动创建的）下。再修改post-update文件内容即可。

可使用的命令：scp root@10.77.144.190:/home/jenkins/workspace/jobs/git\_resource/workspace/RESOUCE/template/git/hooks/\* ./ (当前在新建的hooks目录下)



* + **在项目的gitlab上找到该项目对应的git仓储下的hooks目录，如果该目录是个普通文件夹，那么直接修改post-update即可。**

## 2.3 添加手动打包发布流程

### a) 新建

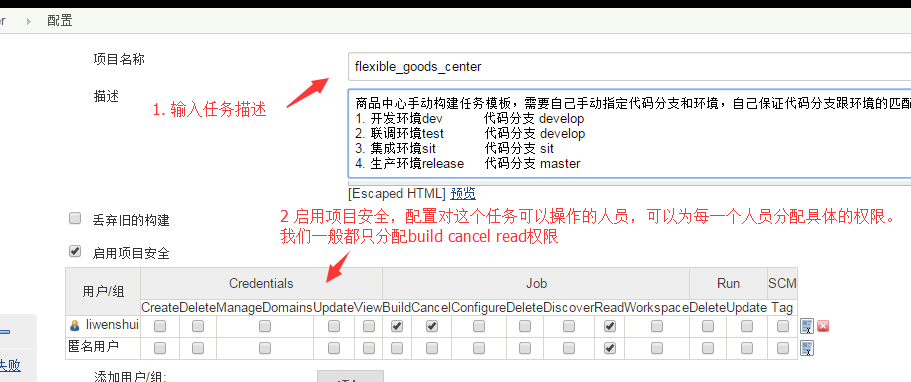
点击左侧的“新建”进行增加CI流程操作（如果没有该功能，请联系Jenkins管理员申请）。(以商品中心为例)



### b) 修改配置

点击“OK”按钮，进入修改配置界面

1. 任务构建策略，权限配置（序号1,2需要修改）



（默认加一个匿名用户，分配读权限。不要问为什么，照做就行）

2.参数化配置（不用修改）



3. 配置动态根据分支下载代码和指定pom文件（序号1,2需要修改）



1. 配置编译后shell脚本（序号1-5需要修改）针对War包的



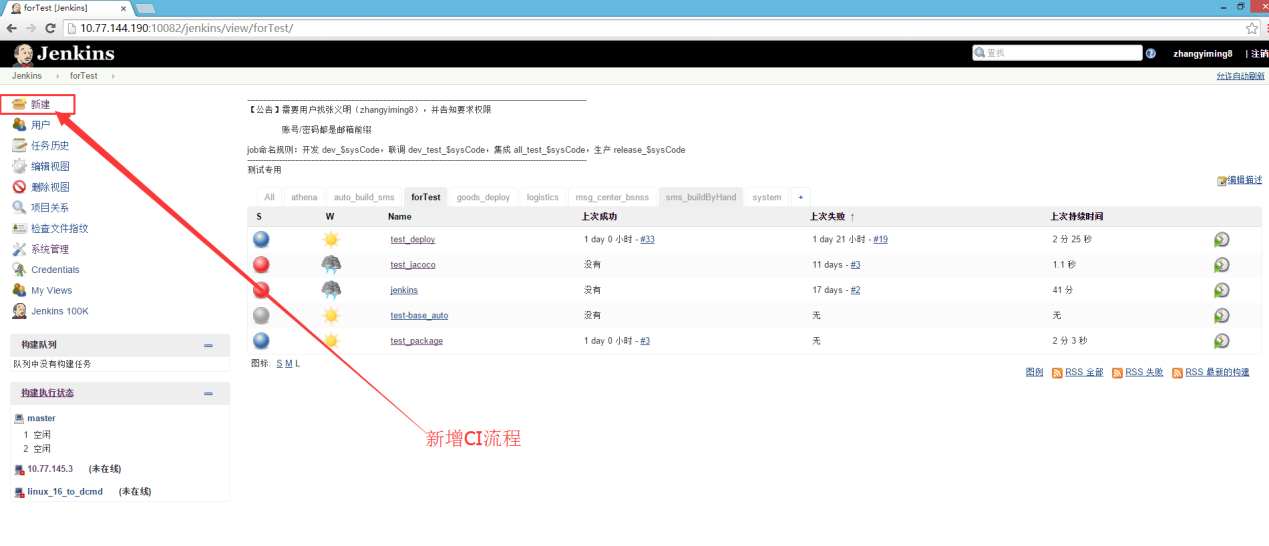
### c) 保存

配置完毕后点击“保存”即可开始构建

## 2.4 添加手动打包发布流程(框架组使用)

### a) 新建

点击左侧的“新建”进行增加CI流程操作（如果没有该功能，请联系Jenkins管理员申请）。



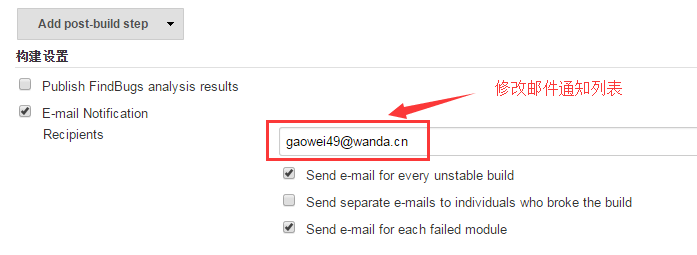
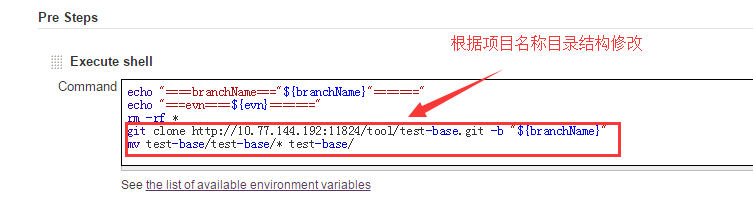
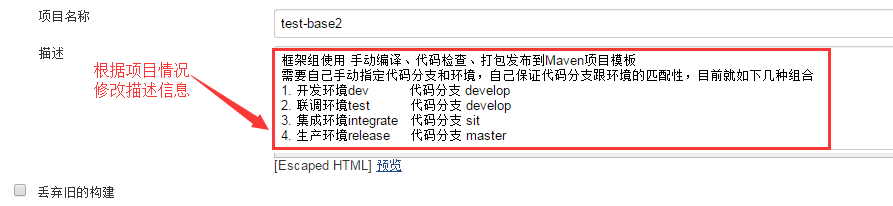
### b) 选择“复制已有的Item”

在“要复制的任务名称”项目中输入”templat\_byhandv2”，输入过程中，系统会在下拉框中给与提示，可以在下拉框中选择自己需要的项目即可。



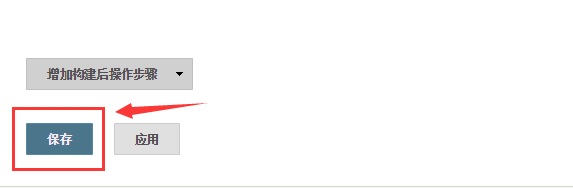
### c) 修改配置

点击“OK”按钮，进入修改配置界面



### d) 保存

配置完毕后点击“保存”即可开始构建



注意事项：如果当前项目的Release版本在Maven仓库上已经存在，构建时可能会失败。

请确认项目pom.xml 定义的版本（version）高于Maven仓库上最新的版本。