# 自动化测试环境搭建

智能产品研究部 罗水波/Robert

## 目录部分

目录

[自动化测试环境搭建 1](#_Toc495589185)

[(一) 目录部分 1](#_Toc495589186)

[(二) software-auto-testing 2](#_Toc495589187)

[一、 用例图 2](#_Toc495589188)

[二、 使用环境资源信息 3](#_Toc495589189)

[三、 使用xshell命令终端连接Centos7.1 5](#_Toc495589190)

[四、 使用xｆｔｐ进行ｗｉｎｄｏｗｓ和Ｌｉｎｕｘ之间的文件传输 6](#_Toc495589191)

[五、 安装Git 8](#_Toc495589192)

[六、 gerrit部署 9](#_Toc495589193)

[七、 Jenkins部署 16](#_Toc495589194)

[八、 jenkins配置——Gerrit tigger 23](#_Toc495589195)

[九、 Jenkins配置——邮件发送 31](#_Toc495589196)

[十、 Jenkins配置——添加slave 33](#_Toc495589197)

[十一、 Jenkins配置——添加slave--实战gsmp\_fw\_slave 37](#_Toc495589198)

[十二、 gerrit配置---创建项目 42](#_Toc495589199)

[十三、 gerrit配置---分组 43](#_Toc495589200)

[十四、 gerrit配置---访问权限access 44](#_Toc495589201)

[十五、 gerrit配置---[verify][code review] 48](#_Toc495589202)

[(三) hardware-auto-testing 48](#_Toc495589203)

[一、 用例图 48](#_Toc495589204)

[二、 环境资源使用信息 48](#_Toc495589205)

[三、 原理图 49](#_Toc495589206)

[(四) FAQs常见问题解答 50](#_Toc495589207)

[一、 Jenkins的admin用户不能登陆了，怎么办？ 50](#_Toc495589208)

[二、 Jenkins slave 没有java web start这个启动项 51](#_Toc495589209)

[三、 为什么要用gerrit来进行code review ? 51](#_Toc495589210)

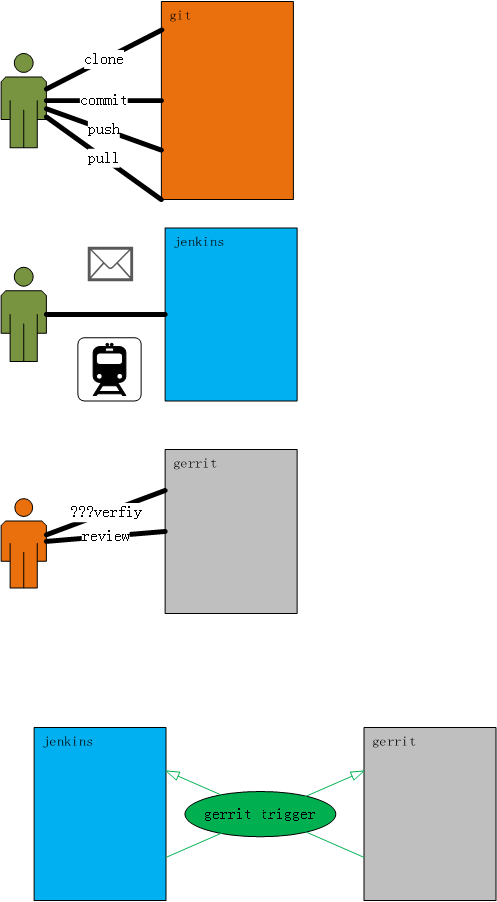
[四、 由于主机IP地址更换,安装在jenkins中的gerrit trigger插件不能触发怎么办？ 52](#_Toc495589211)

[五、 虚拟机在vmware中进行还原后，IP不一致怎么办？ 53](#_Toc495589212)

[六、 这个文档太长，太难掌握，找不到想要的答案，怎么办？ 55](#_Toc495589213)

## software-auto-testing

### 用例图



这一部分是阐述软件自动化测试系统的搭建.其中主要的子系统有git,gerrit,jenkins。在研发过程中采用这套系统，管理人员能够有效的核实开发代码和文档的质量，项目组的各个成员之间也能通过该系统对项目进展有更好的沟通和分析。对于企业来说，这套系统用的都是开源工具，都是免费的，无须支付任何版权和认证费用。

上面的用例图有4个子用例，按照从上到下的顺序这里做下简要说明，

1. 开发人员在git(版本管理工具)上进行操作
2. 系统管理人员通过jenkins进行邮件收发和图形化界面的设置。打个比方，jenkins就像火车 ，可以使人可以更快更方便的到达其他地方
3. 代码复核人员通过gerrit(代码复核工具)进行review和verify。(其中verfiy已经由jenkins自动执行，无需人工执行)
4. Jenkins 和 gerrit 之间 的操作和通信 都是 有gerrit trigger这个插件来完成。

### 使用环境资源信息

#### 电脑IP ，操作系统版本，用途

|  |  |  |  |
| --- | --- | --- | --- |
| 主机 | 操作系统 | IP地址 | 用途 |
| Server-test | Windows server 2012  **(物理主机)** | 192.168.80.156(动态)  还需要wifi连接外网 | 部属vmware 密码：china@1949 |
| server02 | Centos7.1(虚拟机) | 192.168.80.247(静态)  MAC[08:00:27:8e:bf:03]  (已和IT部申请邦定mac地址) | 部署Git+Jenkins+gerrit+samba  [mysql 5.6]  [openjdk1.8] |
| test | Windows7 (物理机) | 192.168.80.85(动态) | 部署jenkins\_slave  密码：china@1949 |
| Mysql账号 |  | root: 123456 gerrit: 123456 |  |
| 虚拟机账号 | VM-Linux | root : china@1949 |  |
| Jenkins账号 |  | jenkins: 123456 admin:123456 |  |

#### 虚拟机Jdk的版本要选择1.8 .jenkins-slave的jdk版本也要选择jdk1.8

* 虚拟机参考命令

#####Begin centos7.1 yum 安装openjdk###############

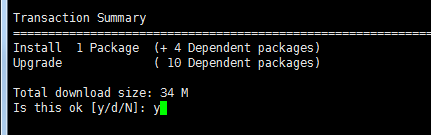
[root@localhost ~]# yum search java|grep jdk

java-1.8.0-openjdk.x86\_64 : OpenJDK Runtime Environment

##2.选择版本,进行安装

//选择1.8版本进行安装

[root@localhost ~]# yum install java-1..0-openjdk



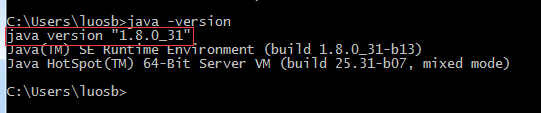
//安装完之后，默认的安装目录是在: /usr/lib/jvm/java-1.8.0-openjdk.x86\_64

#####End centos7.1 yum 安装openjdk###############

更多参考： <http://www.linuxidc.com/Linux/2016-09/134941.htm>

* Jenkins-slave的也可以参考如下图：

在windows 的命令行中输入java -version



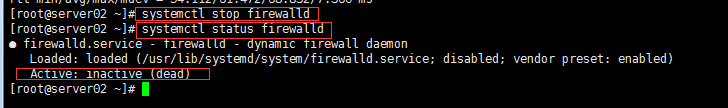
#### 关闭虚拟机的防火墙

参考命令 systemctl stop firewalld 关闭防火墙

systemctl status firewalld 显示防火墙状态

systemctl disable firewalld 禁用防火墙状态

如图所示显示 inactive表示未启动.



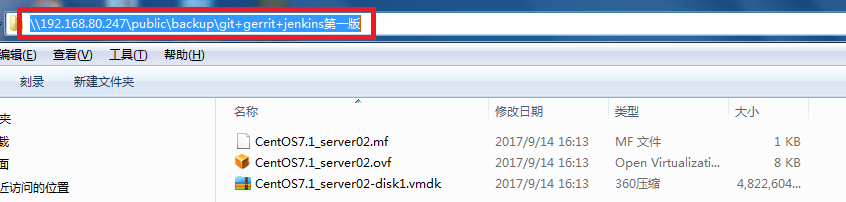
#### 内网资源，工具安装包下载 地址

[\\192.168.80.247\tools](file:///\\192.168.80.247\tools)

\\192.168.80.247\public

* Centos7.1虚拟机备份文件

\\192.168.80.247\public\backup\git+gerrit+jenkins第一版



把git+gerrit+jenkins第一版的文件夹拷贝到windows server 2012服务器上面,再用Vmware workstation 来加载该虚拟机.

第一版是初始版本，只有一个testadmin工程

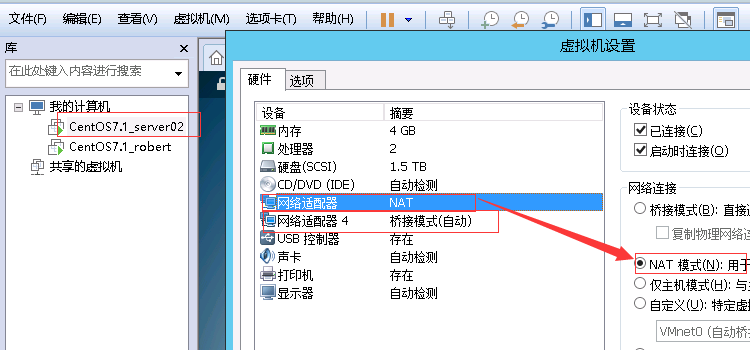
第二版是完整版本，除了testadmin工程还有其他需要使用的工程

#### VM WARE对虚拟机的网络设置如下图

一张网卡是桥接 (有线网卡)

一张网卡是NAT模式 (无线网卡)

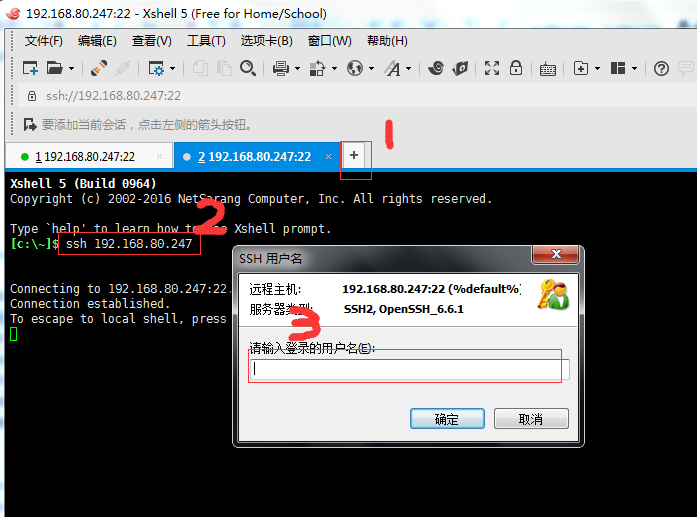
这样才能确保虚拟机能同时用有线网卡上内网和NAT模式上外网。



### 使用xshell命令终端连接Centos7.1

安装Xshell是为了可以远程登录Centos7.1，方便我们随时随地操作Centos7.1

* 使用 Xshell或者secureCRT远程进行Centos7.1的操作请继续看下图.直接通过UI界面进入centos7.1系统的可以跳过此步骤。



安装好之后可以按照图中的步骤进行操作，这样就可以打开一个新的会话来连接到Centos7.1中了.

第一步 ：新建一个选项卡

第二步 : 用 [ssh 192.168.80.247 ] 命令来建立与Centos7.1 的会话

第三步 : 在探出的新窗口输入用户名

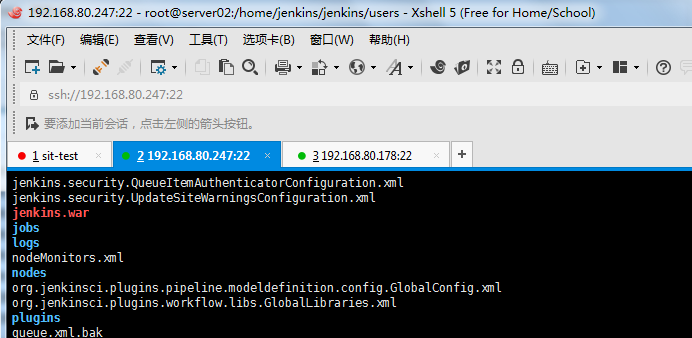
第四部 ：确定再继续输入密码

### 使用xｆｔｐ进行ｗｉｎｄｏｗｓ和Ｌｉｎｕｘ之间的文件传输

#### 下载xftp安装包

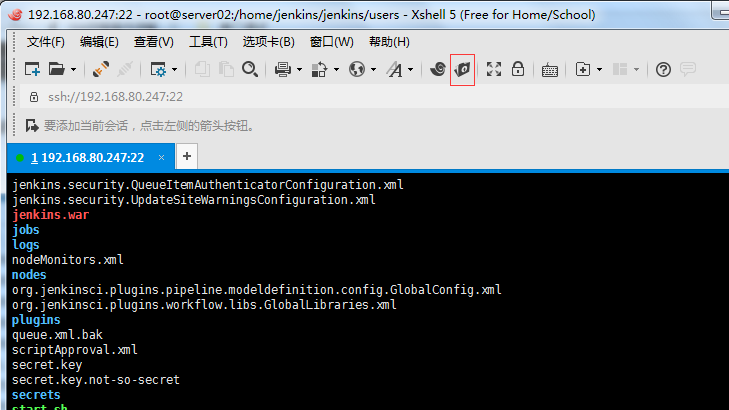
内网下载地址　\\192.168.80.247\public\sf-auto

#### 安装完之后，先用xshell登录

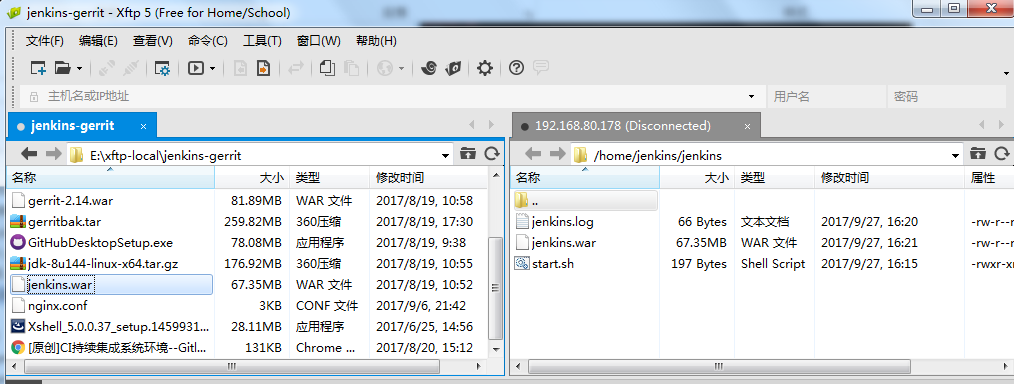


假如你不会用ｘｓｈｅｌｌ那可以参考这里文档的[使用xshell命令终端连接Centos7.1](#_使用xshell命令终端连接Centos7.1)

#### 然后就可以快速用xftp连接上了



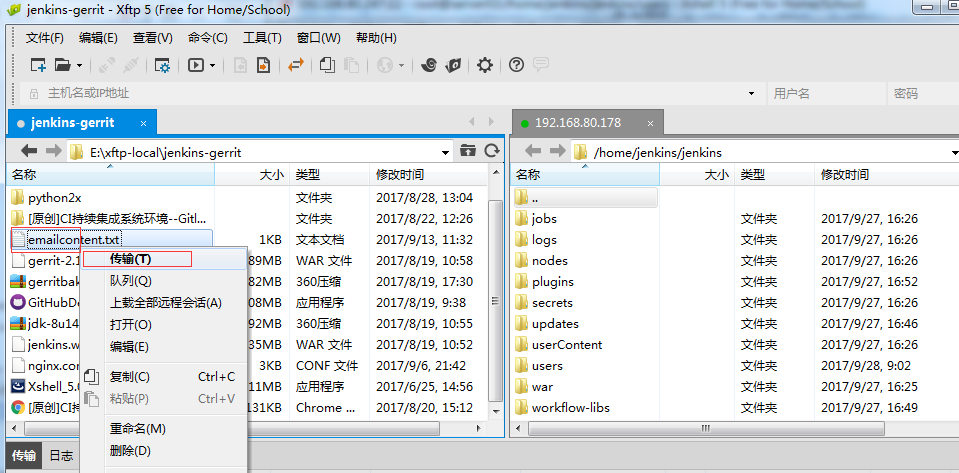
上图中那个红框框内的图标就是xftp的启动图标



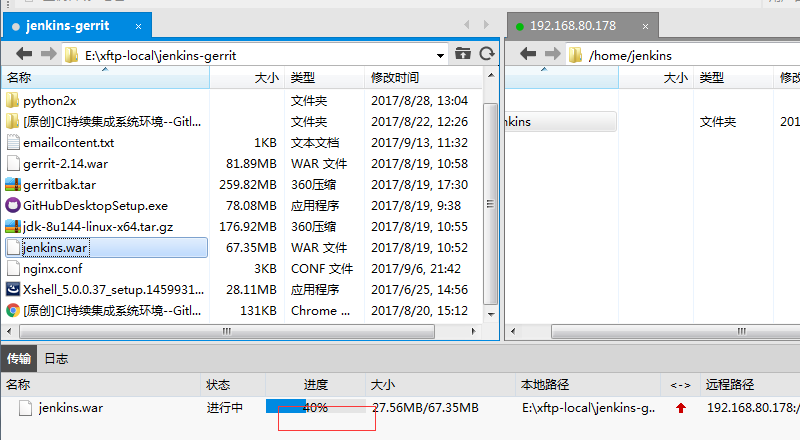
#### 用xftp传输文件

比如我们要把ｗｉｎｄｏｗｓ系统里的文件传输到ｌｉｎｕｘ，我们可以这么来操作　。

首先在左边选中文件，然后鼠标右键有个传输的功能，点击［传输］就开始传输了．



底部的传输进度条显示传输进度４０％，可以明显看出正在传输



其他ｘｆｔｐ的参考

：<http://jingyan.baidu.com/article/a3f121e4b14374fc9152bb5c.html>

### 安装Git

在服务器上部署Git 很简单。不过要先进入 Centos7.1系统。

这里的git是一个命令行模式版本管理工具，

* 使用 Xshell或者secureCRT远程进行Centos7.1 .该方法已在前面有说明请查看

[使用xshell命令终端连接Centos7.1](#_使用xshell命令终端连接Centos7.1)

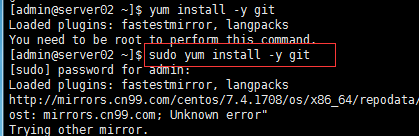
第二种方法就是直接通过UI界面进入centos7.1系统。

* 这Centos7.1中默认已安装 git .



输入命令 [git --version]可以看到 git 的版本号是 git version 1.8.3.1

* 如果需要更新，参考命令[sudo yum install –y git]



使用yum 需要root权限 ，所以这里加了 sudo 。或者你通过root帐户来执行这条命令。

* 关于yum的介绍

Yum（全称为 Yellow dog Updater, Modified）是一个在Fedora和RedHat以及CentOS中的[Shell](https://baike.baidu.com/item/Shell)前端软件包管理器。基于[RPM](https://baike.baidu.com/item/RPM)包管理，能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软件包，无须繁琐地一次次下载、安装

### gerrit部署

#### 准备安装gerrit—安装mysql5.6

备注一下：gerrit自带H2数据库的不太适合后期维护，我们这里选择很流行的开源数据库mysql.安装好Mysql后，我们可以用如下命令来操作

* *#####Begin centos7.1安装mysql5.6###############*

*##检查linux版本*

*[root@server ~]$* ***cat /etc/redhat-release***

*CentOS Linux release 7.1.1503 (Core)*

*##加入yum的repo*

*[root@iZ28gvqe4biZ ~]# rpm -Uvh http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm*

*##yum的repo开启mysql*

*[root@iZ28gvqe4biZ ~]# yum repolist enabled | grep "mysql.\*-community.\*"*

*##在线安装mysql [如果下载慢，可以取消重新下载再安装]*

*[root@iZ28gvqe4biZ ~]# yum -y install mysql-community-server*

*##允许mysqld服务启动*

*[root@iZ28gvqe4biZ ~]# systemctl enable mysqld*

*##启动mysqld服务*

*[root@iZ28gvqe4biZ ~]# systemctl start mysqld*

*##重置密码*

*[root@iZ28gvqe4biZ ~]# mysql\_secure\_installation*

*##创建数据库实例reviewdb设置字符集*

**以下命令也可用于重新安装数据库实例**

*###创建数据库用户gerrit[先会删除用户gerrit]*

*mysql> drop user gerrit;*

*mysql> Delete FROM user Where User=' gerrit' and Host='localhost';*

*mysql>flush privileges;*

*###创建数据库reviewdb[先会删除数据库reviewdb*

*mysql>drop database reviewdb; ##//删除用户的数据库*

*mysql> create database reviewdb DEFAULT CHARSET utf8 COLLATE utf8\_general\_ci;*

*mysql> create user ' gerrit'@'%' identified by '123456';*

*###给用户admin赋权限[本地连接]+[远程连接]*

*mysql> grant all privileges on `reviewdb`.\* to ' gerrit'@'%' identified by '123456';*

*mysql> grant all privileges on `reviewdb`.\* to ' gerrit'@'localhost' identified by '123456';*

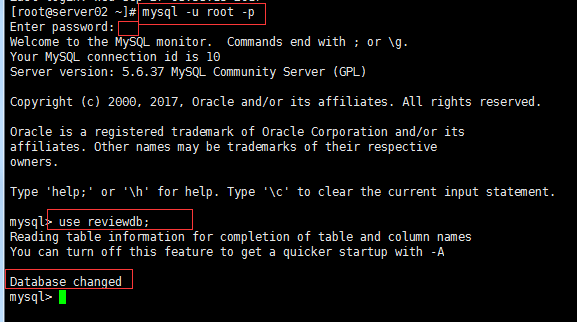
*###更新权限*

*mysql> flush privileges;*

*##参考网址 http://www.cnblogs.com/XBlack/p/5178758.html*

*#####End centos7.1安装mysql5.6###############*

* 验证一下mysql 是否安装好。



如上图所示，

* 1. 首先进入Centos7.1的命令行，输入命令 [mysql –u root –p ]
  2. 然后输入mysql 数据库 root用户的密码123456
  3. 输入[use reviewdb]，进入review数据库实例
  4. 看到提示数据库切换成功，看到这个那就恭喜你Mysql数据安装成功，并且成功创 建gerrit所需要的数据库实例reviewdb。

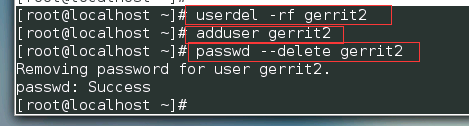
#### 准备安装gerrit—增加gerrit2用户

Gerrit server是部署在虚拟机centos7.1操作系统之上的。

[userdel -rf gerrit2] 命令是删除 gerrit2 用户和所在目录

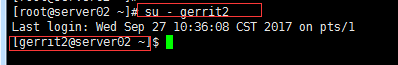
[adduser gerrit2] 命令是增加gerrit2 用户

[passwd --delete gerrit2] 命令是设定gerrit2用户不需要使用密码登录



验证一下gerrit2用户是否可以使用

使用 [su – gerrit2 ]命令可以看到切换用户成功。” [gerrit2@server02 ~]”表示切换到gerrit2的用户目录

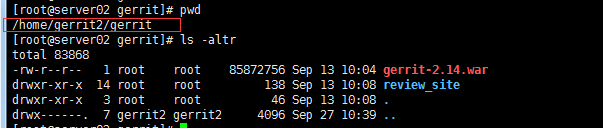


* 更多关于Linux增加删除用户的命令可以参考

<http://www.cnblogs.com/DaDaOnline/p/5527833.html>

#### 安装gerrit

工作目录是/home/gerrit2/gerrit,我们需要先进入此目录



Gerrit的安装步骤基本可以默认回车键，这里要拿出来说明的是几点.

* Database server type 要选择我们前面已经装备好的mysql，然后数据库的用户名，密码根据前面设定的使用gerrit2 , 123456 .
* Authentication method 选择DEVELOPMENT\_BECOME\_ANY\_ACCOUNT

这是一种开发者的模式，比较方便。如果正式使用再更换成HTTP模式.

* HTTP Daemon 里要配置Listen on port 和 Canonical URL
* Label Verify 要安装，这是后期jenkins需要的一个操作。 [高版本gerrit 2.14.0以上]

*#####Begin centos7.1安装gerrit 2.14###############*

*#1 各个版本下载链接*

*http://gerrit-releases.storage.googleapis.com/index.html*

*#2 公司内网下载地址*

[*\\192.168.80.247\public\sf-auto*](file:///\\192.168.80.247\public\sf-auto)

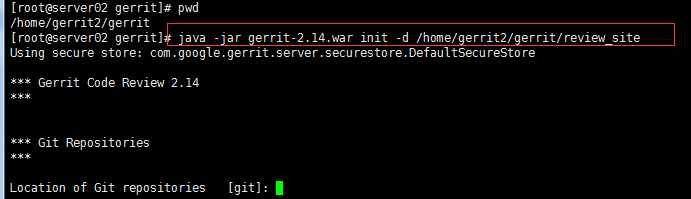
* *##进入/home/gerrit2目录，输入以下命令安装*

*java -jar gerrit-2.14.war init -d /home/gerrit2/gerrit/review\_site*

参考说明:

java –jar是启动java虚拟机的作用

init –d 是初始化的目录的作用



*安装要修改的地方*

*\*\*\*Database server type [mysql]:*

*Database name [reviewdb]:*

*Database username [admin]:*

*admin's password :*

*confirm password :*

*\*\*\*Authentication method [DEVELOPMENT\_BECOME\_ANY\_ACCOUNT]:*

*\*\*\* HTTP Daemon*

*\*\*\**

*Behind reverse proxy [y/N]?*

*Use SSL (https://) [y/N]?*

*Listen on address [\*]:*

*Listen on port [8080]: 8999*

*Canonical URL [http://localhost:8999/]: http://192.168.80.247:8999/*

*\*\*\* Gerrit Administrator*

*Create administrator user [Y/n]? y*

*username [admin]: admin*

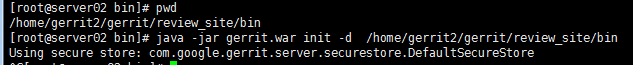
*name [Administrator]:*

*HTTP password [secret]: 123456*

*##The index must be rebuilt before starting Gerrit:*

*cd /home/gerrit2/gerrit/review\_site/bin*

*java -jar gerrit.war init -d /home/gerrit2/gerrit/review\_site/bin*



*##安装git-review*

*yum -y install epel-release*

*yum install python-pip*

*pip install git-review*

*更多关于git-review的介绍* [*https://www.mediawiki.org/wiki/Gerrit/git-review*](https://www.mediawiki.org/wiki/Gerrit/git-review)

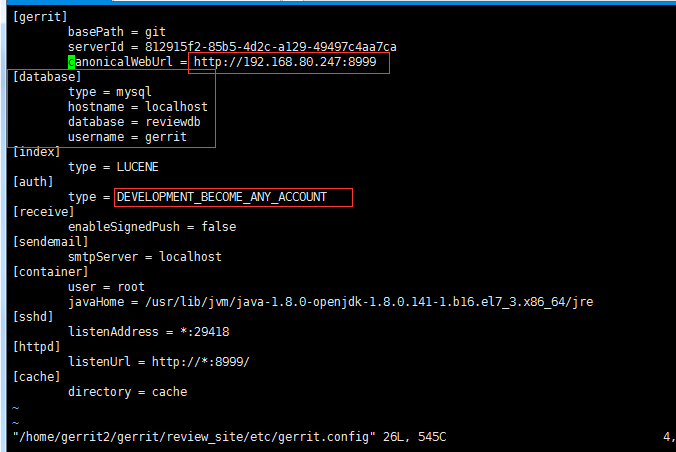
*#####End centos7.1安装gerrit 2.13###############*

#### 验证gerrit是否安装正确

验证一下gerrit的配置文件是否正确。

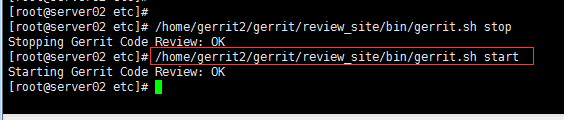
到/home/gerrit2/gerrit/review\_site/etc 目录下打开gerrit.config文件

[cat /home/gerrit2/gerrit/review\_site/etc/gerrit.config]

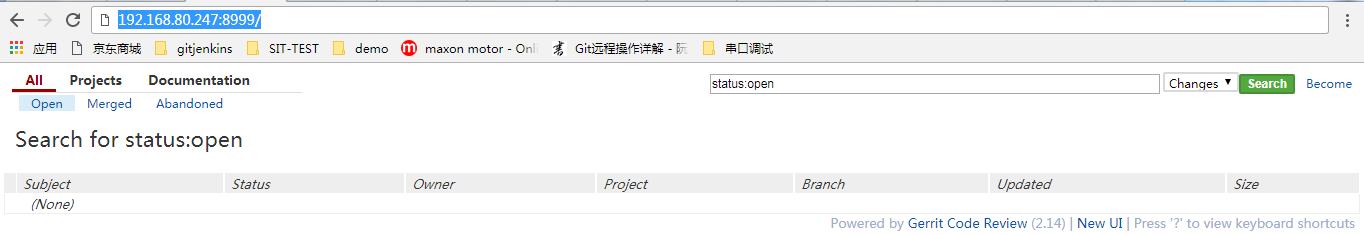


验证一下gerrit的web 服务是否能启动OK。

/home/gerrit2/gerrit/review\_site/bin/gerrit.sh start 命令是启动gerrit服务。当提示Starting Gerrit Code Review: OK 表示启动成功。



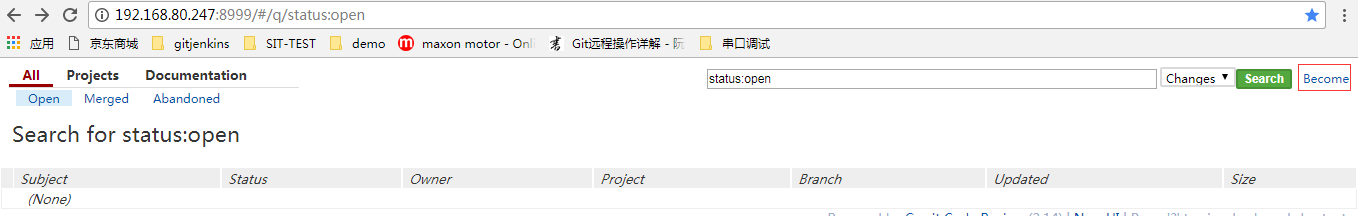
安装成功打开网址 [<http://192.168.80.247:8999/>]会显示如下



#### 添加gerrit的jenkins账户

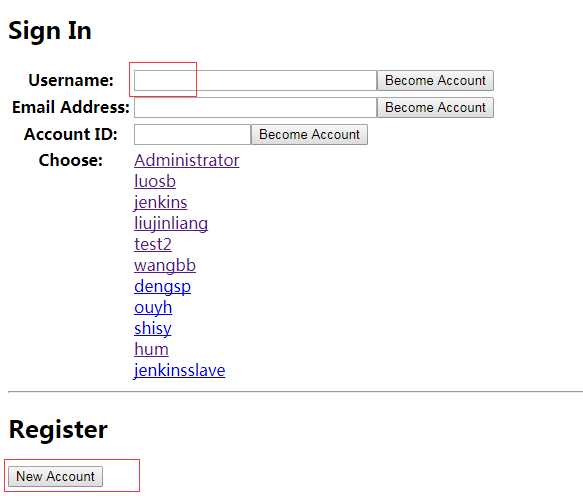
添加这个jenkins账户是为了给在gerrit 中有个jenkins账户可以使用。

##### 进入gerrit的添加账户界面中。



*gerrit右上角有个状态切换的按钮。我们可以点击become ，进入*

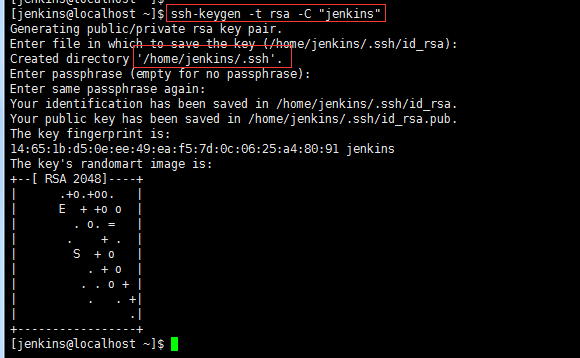
##### 输入Jenkins账户信息



输入username jenkins再点击New Account就可以注册了。

不过，jenkins账户现在还不能远程访问，要在这里要配对一个public key 才能在远端使用。

##### 在Centos7.1系统中生成 ssh-key。



先切换到jenkins用户目录,然后输入 ssh-keygen -t rsa -C "jenkins" 命令生成key .

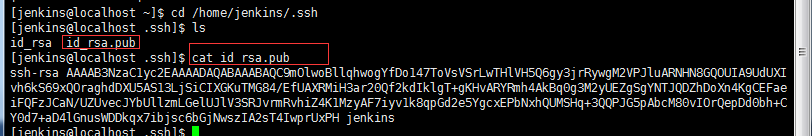
ssh-keygen 是调用ssh-kegen程序

-t rsa 表示加密类型是rsa

-C “jenkins”是注释为jenkins

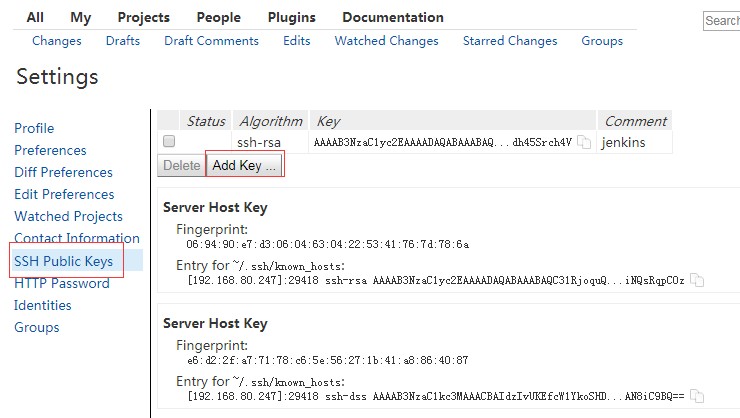
如下图可以看到id\_rsa.pub文件生成成功 。

其中用cat id\_rsa.pub 命令可以看到内容。



##### 在gerrit 中填入ssh-key

注册账户之后，进入设置界面settings，把生成的Key导入.

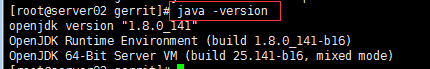


### Jenkins部署

#### 准备安装Jenkins—安装jdk1.8

1. 检查是否已安装Jdk1.8 ，如果没有请参考2的命令

因为我们安装的jenkins版本是2.70.1需要jdk1.8作支持，请确保安装的版本是jdk1.8或者以上。



1. 参考安装命令

#####Begin centos7.1 yum 安装openjdk###############

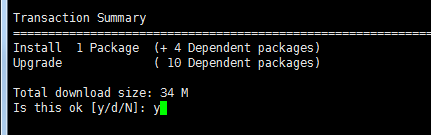
[root@localhost ~]# yum search java|grep jdk

java-1.8.0-openjdk.x86\_64 : OpenJDK Runtime Environment

##2.选择版本,进行安装

//选择1.8版本进行安装

[root@localhost ~]# yum install java-1.8.0-openjdk



当提示是否要下载的时候，输入y 表示确定.

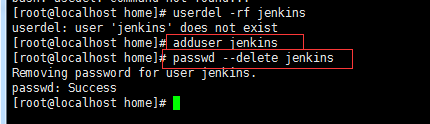
//安装完之后，默认的安装目录是在: /usr/lib/jvm/java-1.8.0-openjdk.x86\_64

#####End centos7.1 yum 安装openjdk###############

更多参考： <http://www.linuxidc.com/Linux/2016-09/134941.htm>

#### 准备安装Jenkins—增加jenkins用户

1. 新建用户adduser jenkins
2. 设定用户jenkins的密码是空



注意：如果需要删除用户可以使用userdel –rf Jenkins 命令

#### 安装Jenkins—

##### 进入工作目录 /home/jenkins/jenkins

如果没有此目录请先创建 mkdir /home/jenkins/jenkins



##### 新建start.sh的启动脚本

vim start.sh

* 温馨提示vim是一个常用的linux工具,参考说明 <http://www.runoob.com/linux/linux-vim.html>

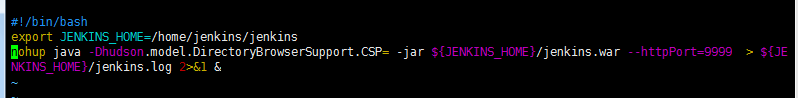
在其中输入如下字符，表示导入jenkins的环境变量，并且用nohup在后台启动

#!/bin/bash

export JENKINS\_HOME=/home/jenkins/jenkins

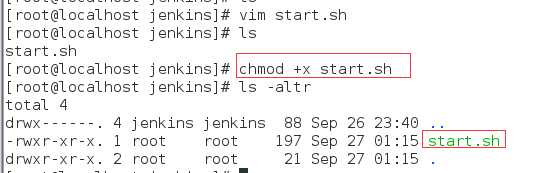
nohup java -Dhudson.model.DirectoryBrowserSupport.CSP= -jar ${JENKINS\_HOME}/jenkins.war --httpPort=9999 > ${JENKINS\_HOME}/jenkins.log 2>&1 &

创建成功如下图:



给start.sh文件增加可执行权限，权限添加成功后会文件名字变成绿色

chmod +x start.sh

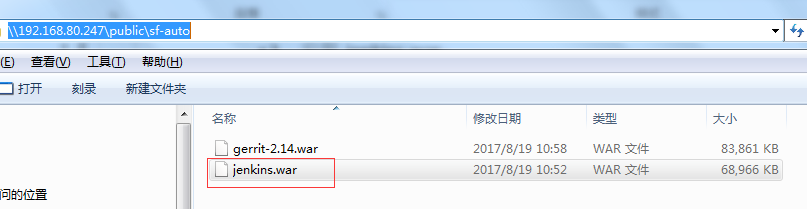


##### 安装jenkins.war

1. 下载安装包，

* 在内网共享目录可以 [\\192.168.80.247\public\sf-auto](file:///\\192.168.80.247\public\sf-auto) ,并上传到/home/jenkins/jenkins。

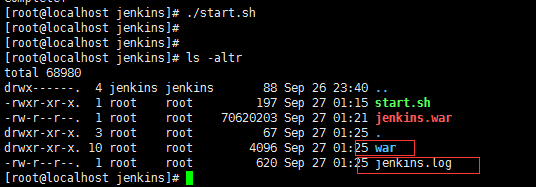
可以用xftp 或者winscp的工具来上传.



* 外网下载地址 ：https://pkg.jenkins.io/redhat-stable/

1. 启动start.sh脚本

会生成war和其他文件.我们用 ls –altr命令来显示当前目录下的所有文件



1. 启动Jenkins后，下载必要的插件

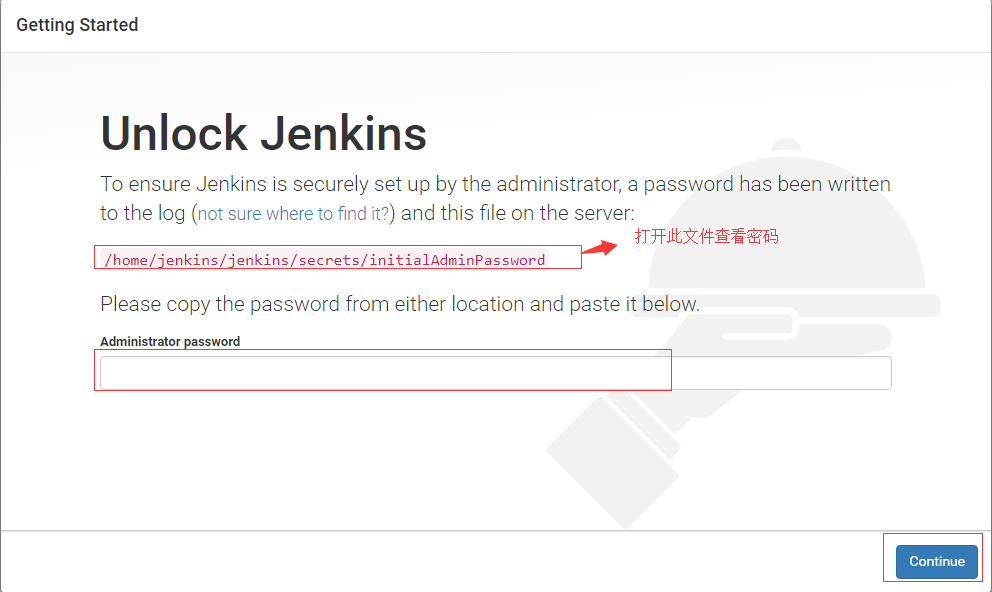
在浏览器中输入<http://192.168.80.178:9999/> 地址，回车键确认，打开web ui 界面

遇到如下页面，请不要惊慌，不要失望。这只是jenkins为了安全，让你输入密钥而已。

我们可以在虚拟机centos7.1中找到文件.

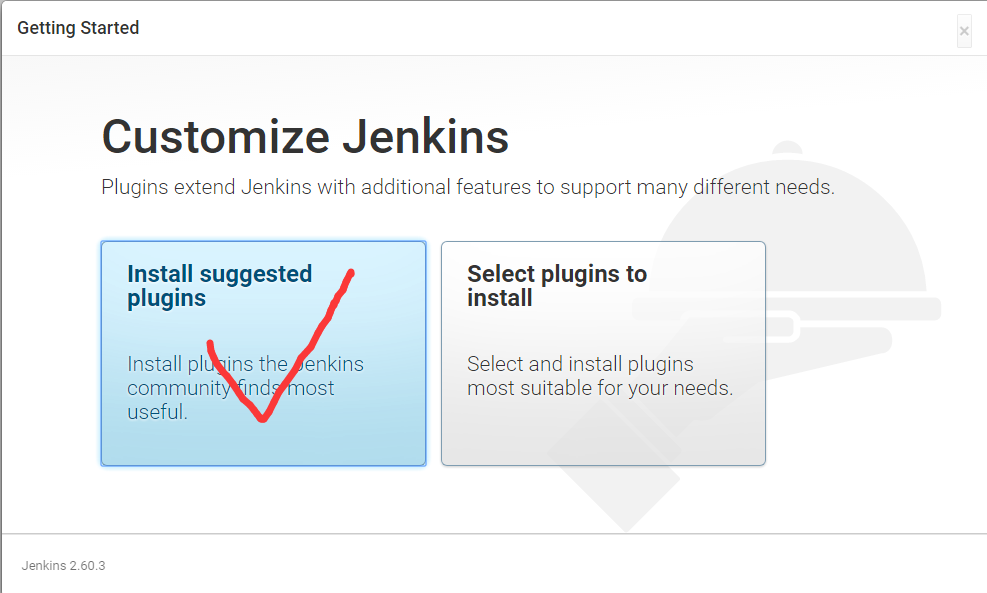
* cat /home/jenkins/jenkins/secrets/initialAdminPassword

把文件里面的字符拷贝到页面的输入框即可

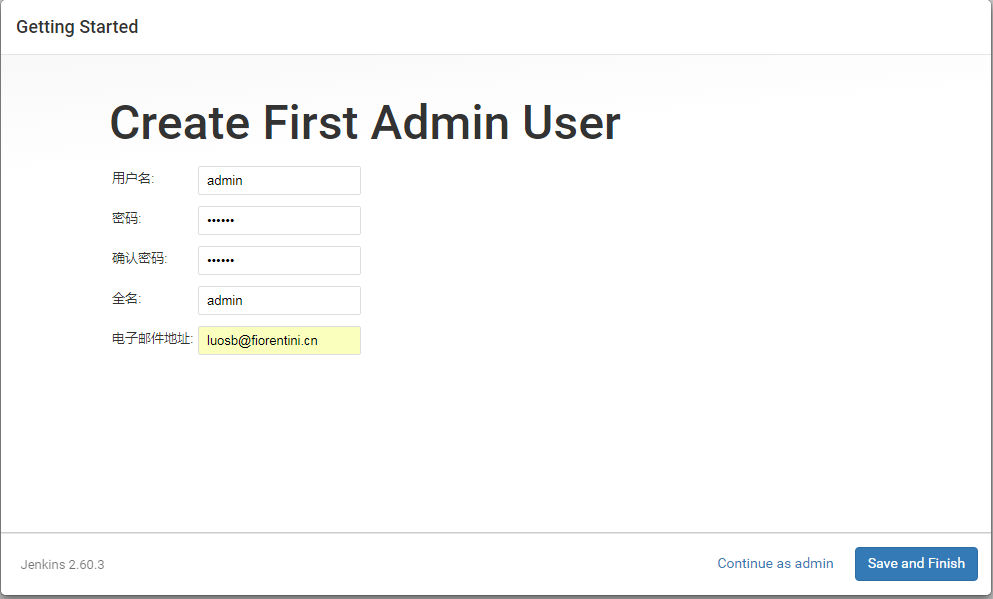


如果此web图形页面没打开，请检查系统防火墙是否关闭.[关闭虚拟机的防火墙](#_关闭虚拟机的防火墙)

* 接下来选择左边的默认安装Plugins



* 输入admin用户名密码，这里记得先输入admin用户 .选择Continue as admin



* 出现这个界面表示安装成功.



#### 验证jenkins安装成功

在浏览器中输入ip地址加端口



#### 增加Jenkins的用户

打开jenkins之后，我们需要立即创建一个账户，以防不备。

这里以账户名 Jenkins ， 密码 123456 为例 。

1. 在ｊｅｎｋｉｎｓ的左侧选择**系统管理**
2. 进入　［**管理用户**　］



1. 再点击［**新建用户**］



1. 输入正确的用户名和密码等信息



### jenkins配置——Gerrit tigger

#### 准备安装插件gerrit trigger

安装插件的主要步骤，笔者简单列一下：

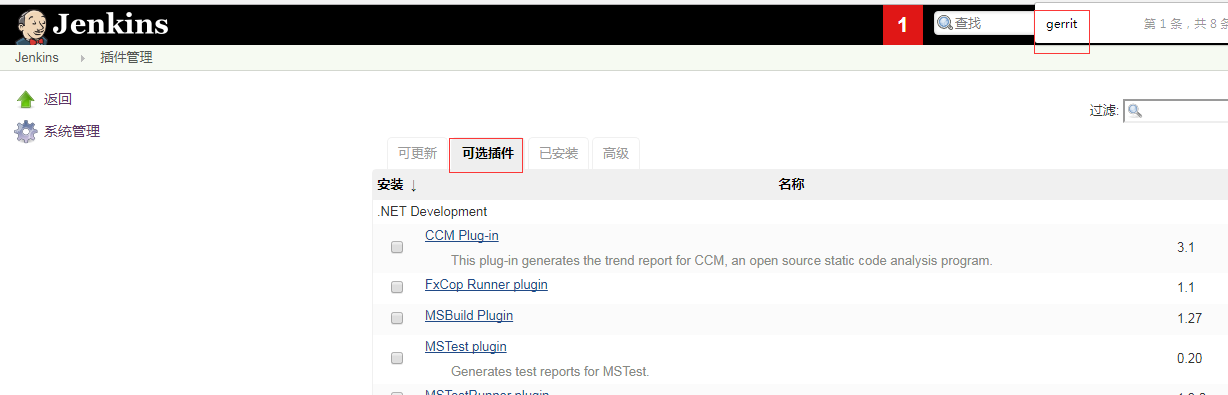
1. [系统管理]---[管理插件]



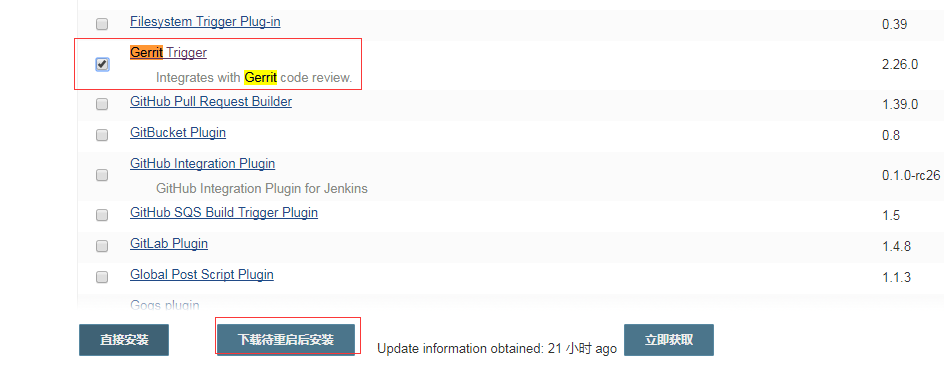
1. [系统管理]---[管理插件]

#### 安装并重启

* 进入可选插件这一栏，然后用ctrl+f 的组合键调用浏览器的查询功能 ，查找gerrit的相关插件



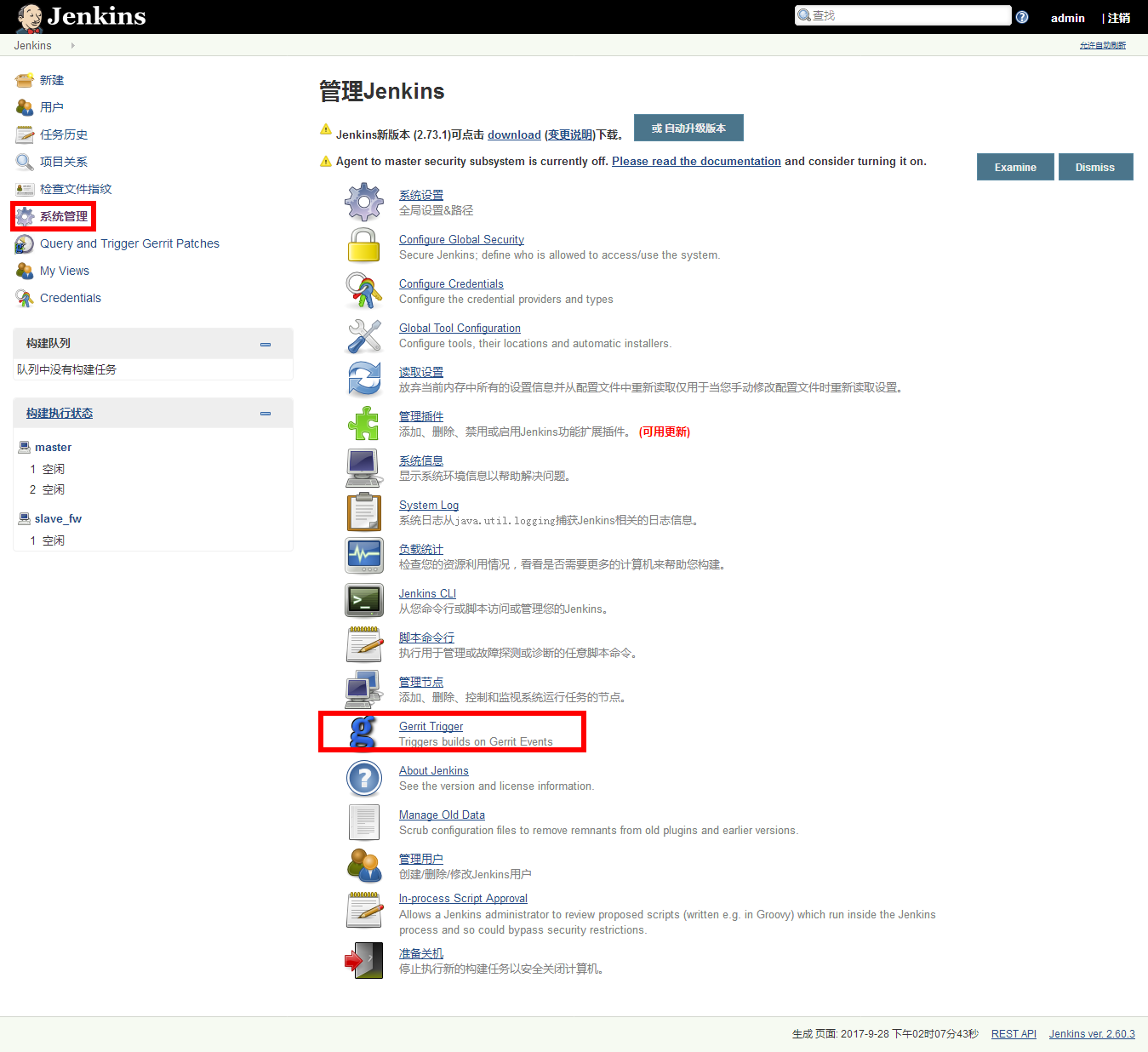
* 可以找到一个Gerrit Trigger的插件，我们这里勾选Gerrit Trigger ,并且选择下方的下载待重启后安装



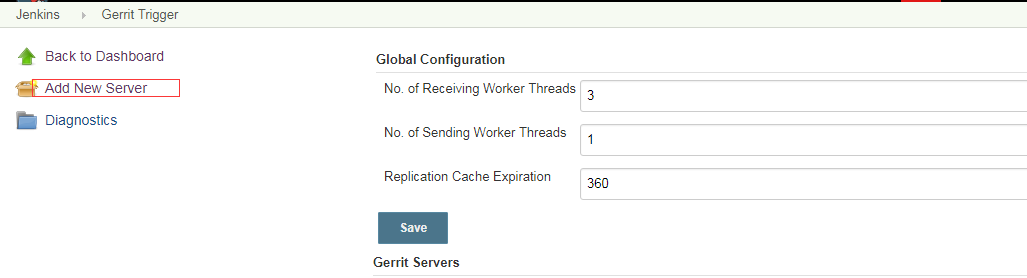
#### 新增gerrit trigger

##### 找到已安装Gerrit Grigger插件

[系统管理]---[Gerrit Trigger]

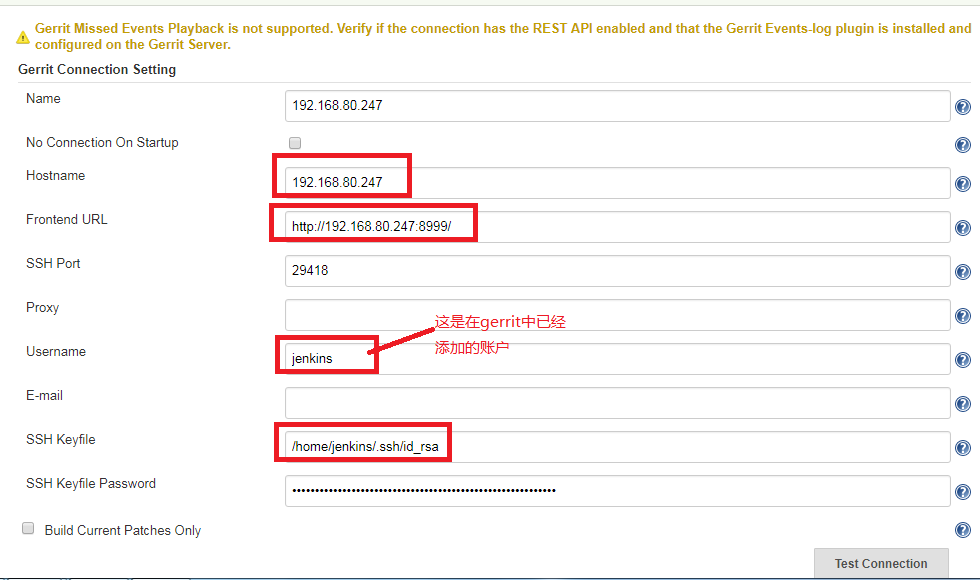


##### 添加 gerrit server



通过上图中的Add New Server 可以添加我们网络中的gerrit server 服务器。

添加的过程中配置的信息如下图所示。要注意[ username] ,[ssh key file ]的填写要准确.如果不知道Jenkins 和SSH KEYFILE从何而来，可以参考文档的前面部分. [准备安装Jenkins—增加jenkins用户](#_准备安装Jenkins—增加jenkins用户) .[在Centos7.1系统中生成 ssh-key。](#_在Centos7.1系统中生成_ssh-key。)

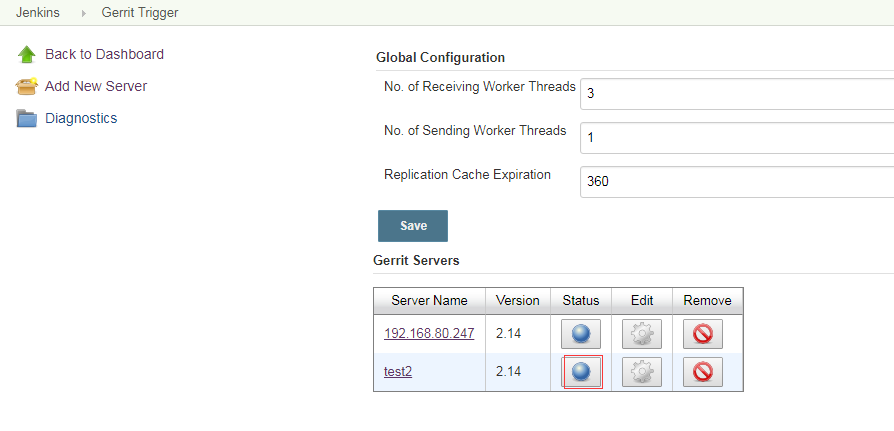


* 点击底部有个Test Connection按钮验证一下是否添加成功。

##### 查看 gerrit server的状态

如下图所示，蓝色表示Status是正常的。注意这个蓝色的图标是可以点击来切换状态的。

加入你想停掉一个gerrit server 就可以点击这个圆形蓝色的图标



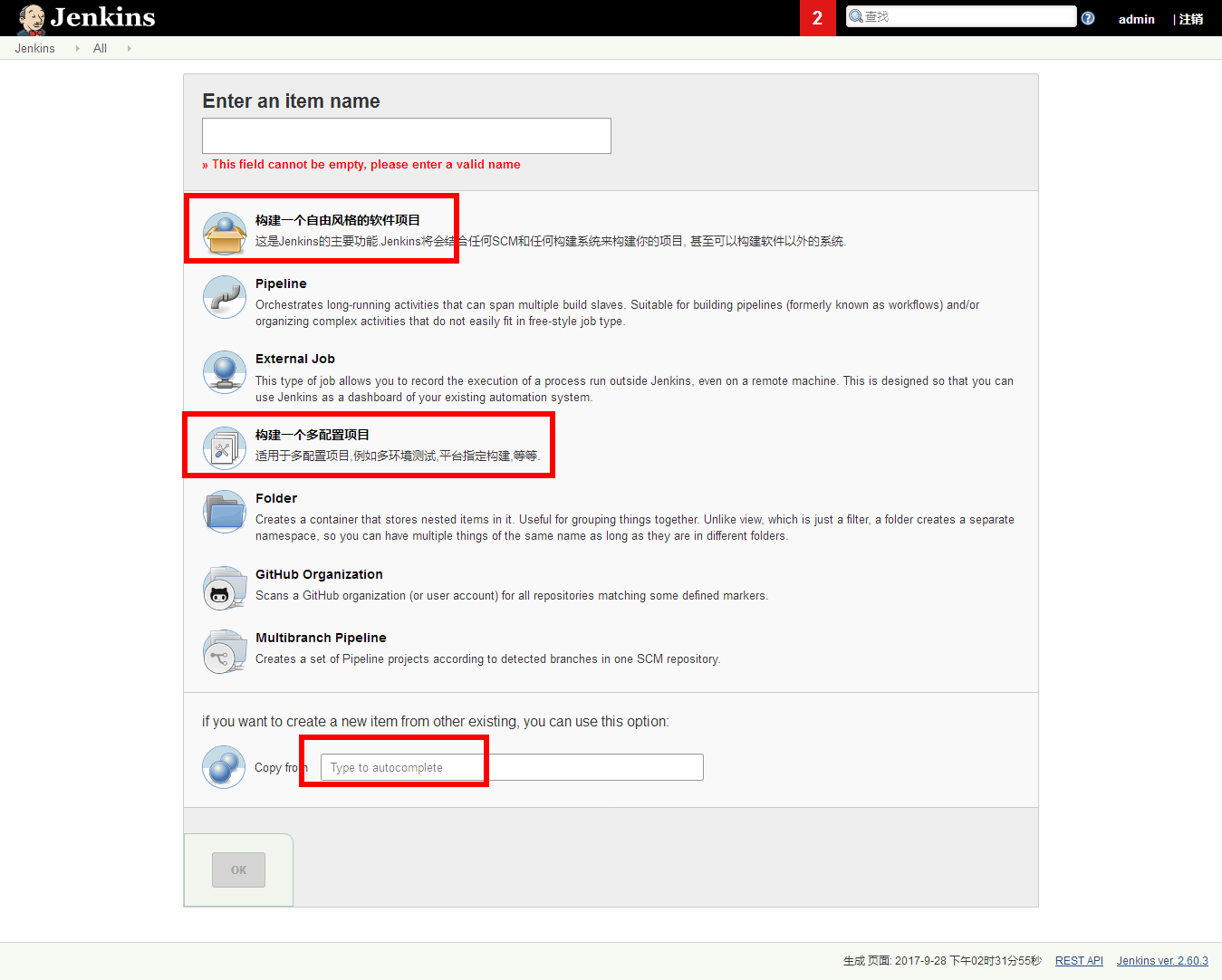
点击成功，状态切换成停止。如下图红色部分



#### 新建jenkins的工程

##### [新建]工程

##### 选择[构建 自由风格的软件项目]



* 这里做下说明，什么时候用[构建一个多配置项目]？

当你需要用到jenkins-slave的时候就需要选择这个项目。比如我们这里固件的冒烟测试需要 在windows 7 上面进行，就要选择 [构建一个多配置项目].置于如何添加jenkins的slave，我们可以参考这里.[Jenkins配置——添加slave](#_Jenkins配置——添加slave)

* [Copy to ]是一个 非常方便的功能 。可以复制一抹一样的jenkins工程。我相信大家肯定能用到 。

#### 配置jenkins中gerrit trigger

##### 进入新建工程的[配置]项



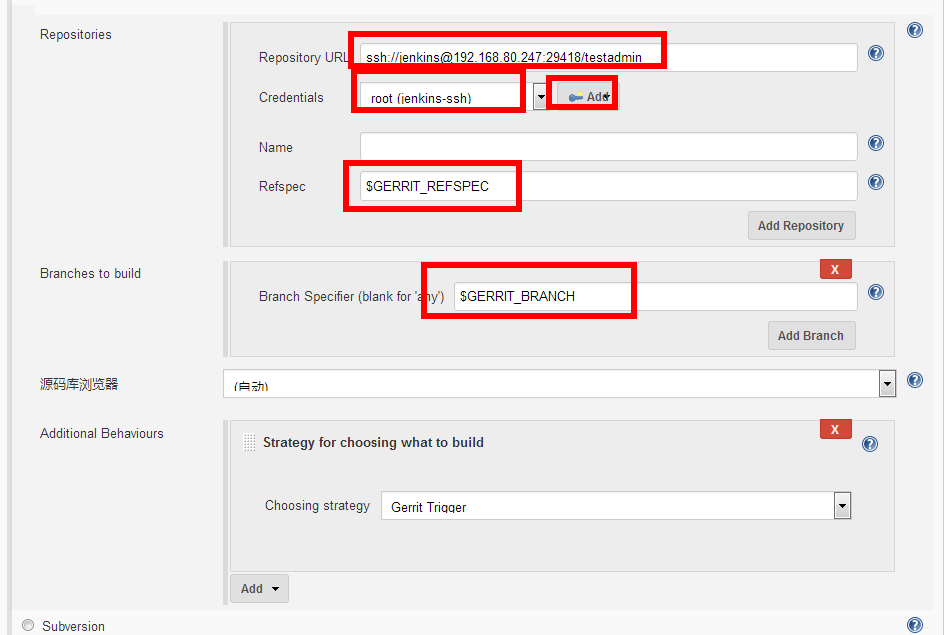
##### 整个工程配置可以参考以下图片和html 文件

参考图片(其中html格式的小部分与需要与服务器通信的内容会被裁掉，无法显示完整。)

或者html文件

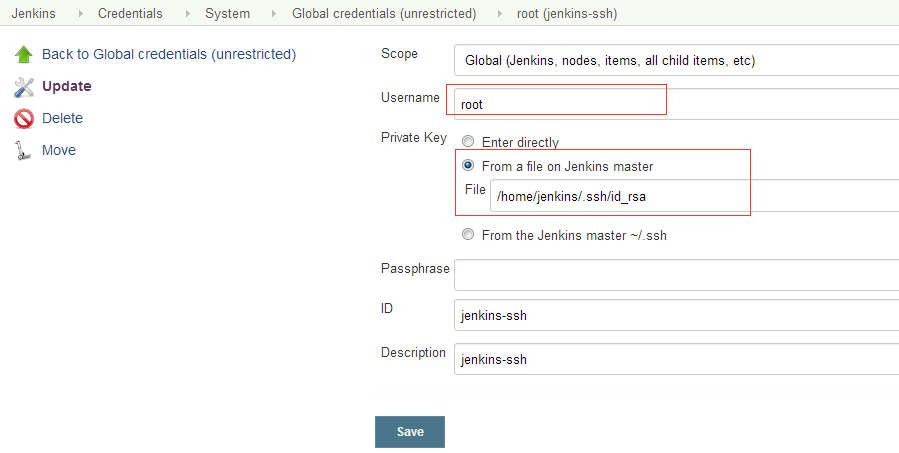
##### 整个工程配置主要有三部分：

1. 源码管理



其中jenkins账户和Credentitials是已经设定好的 .

旁边那个 Add的按钮是用来增加jenkins-ssh路径的，具体ssh-key的Credentitials设置如下图所示。

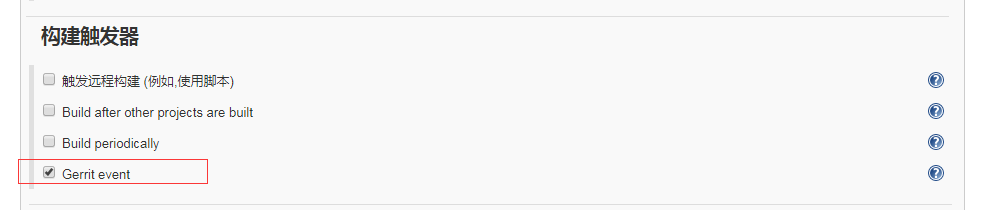


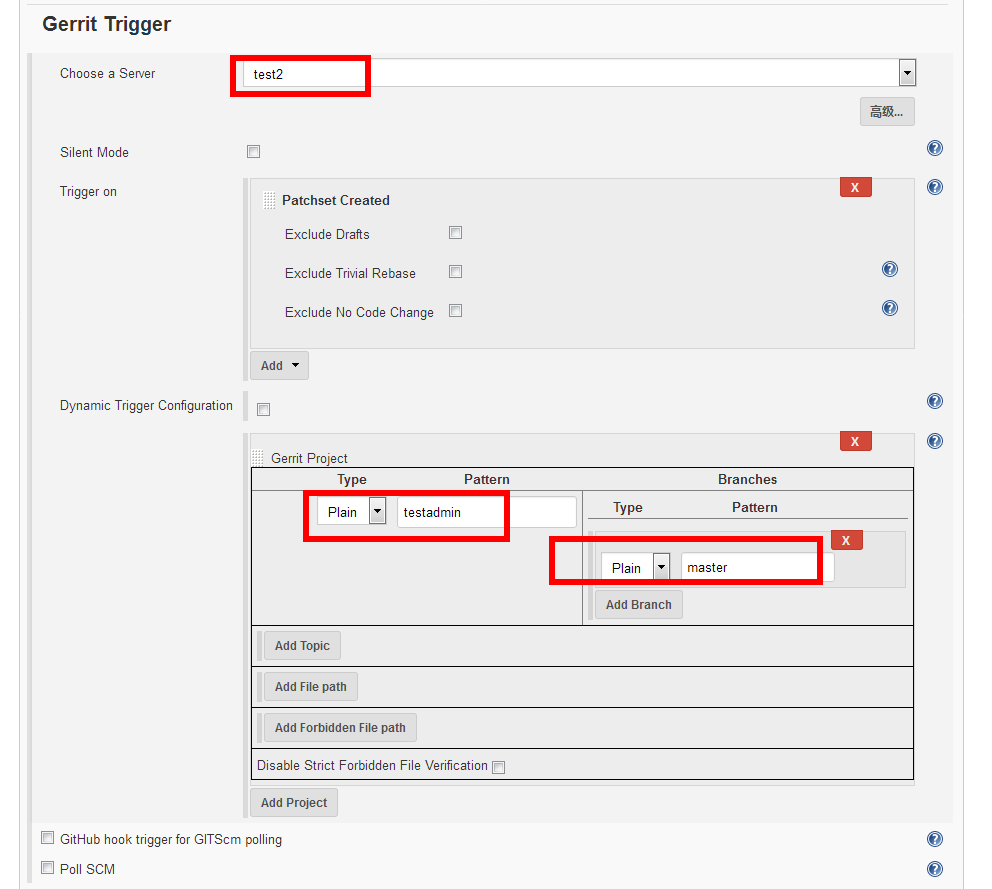
* 有没有理解这里的username为什么是root ? 其实这是因为，gerrit server是我们用Centos7.1的root 用户启动的，所以这里会显示root用户。

1. 触发器构建

在勾选gerrit event 之后，配置正确的选项.

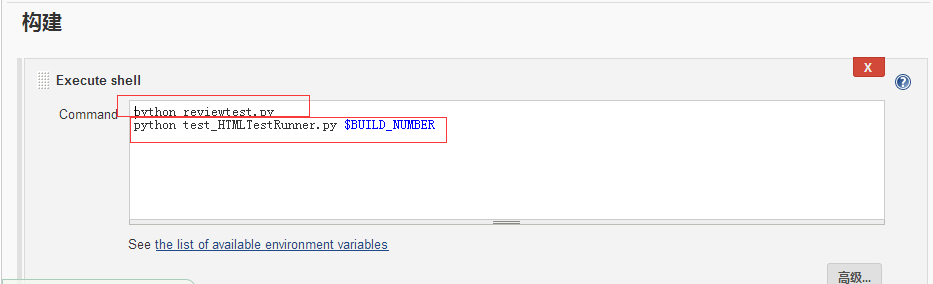
Choose a Server : 选择之前已经添加的服务。 如果忘记了，可以参考[添加 gerrit server](#_添加_gerrit_server)



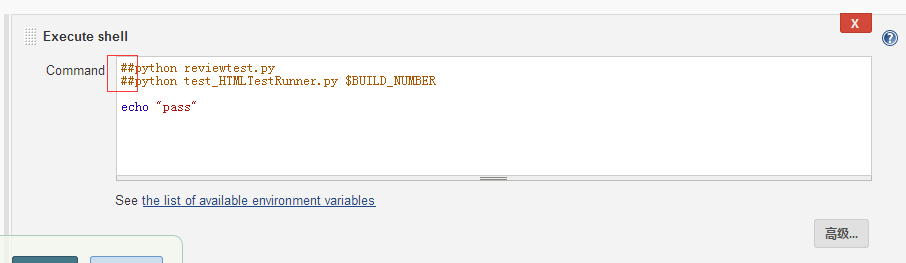


1. 构建

这里是在192.168.80.247这个服务器上面执行两个Python脚本。



[如果你的工程目录下面没有Python 脚本，那么出错是正确的。这时候，你需要让这里直接通过，而不执行这些脚本。如下图 ]



### Jenkins配置——邮件发送

##### 进入邮件设置界面



##### 填写正确的jenkins location



这两个地址根据实际情况要写正确。对以后的影响很大。

而且这个账户在第4步中还要进行验证.

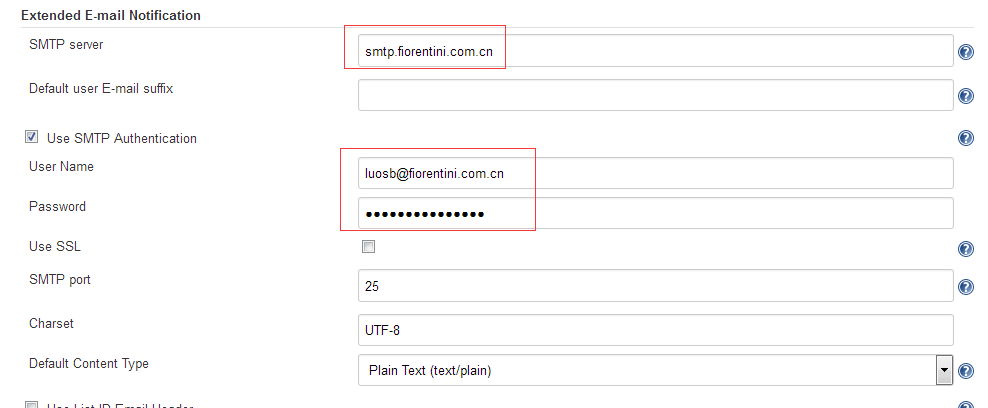
##### 填写正确的Extended E-mail Notification配置项

如果你没有这个选项，那么你需要安装Jenkins的插件[Extended E-mail Notification]

在连上互联网的情况下参考<http://blog.csdn.net/houyefeng/article/details/51344337>。

主要参考如下图。

其中USE SMTP authentication这一项是实际的邮件用户名和地址(luosb的邮箱密码：Luoshuibo123456 )。



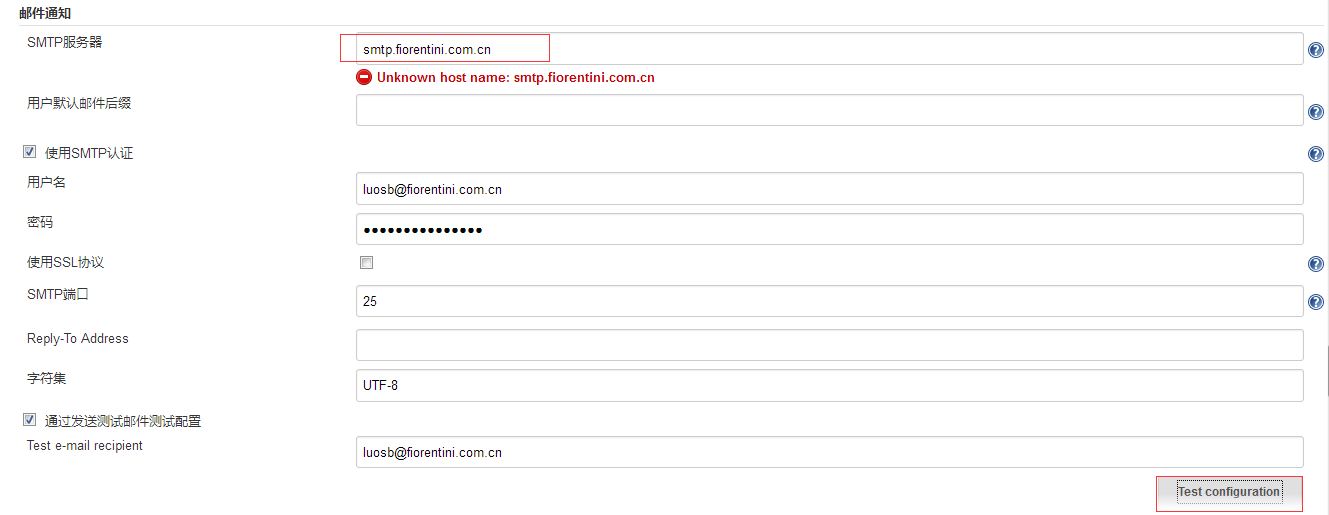
##### 参考 jenkins能发送邮件的配置图和html文件

(其中html格式的小部分与需要与服务器通信的内容会被裁掉，无法显示完整。)



##### 验证jenkins能正确的发送邮件

* 我们首先用Jenkins自带的邮件通知来确认邮件的发送机制正常。只有机制正常，我们才能进一步确定Extended E-mail Notification这个插件也能发送邮件成功。



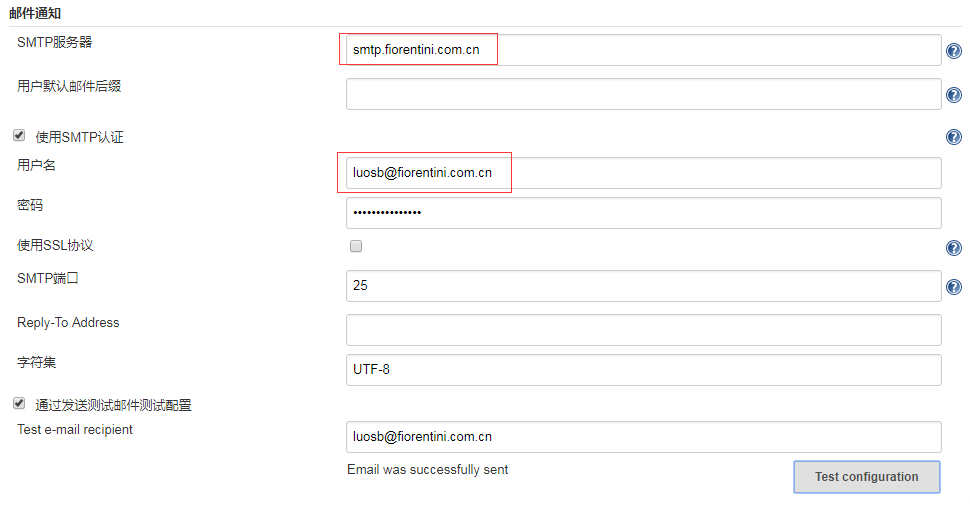
unkown host name .这个红色错误提示意思是找不到smtp服务器，也就是网络通信失败了。你需要确保你的网络中你能Ping通这个SMTP服务器。

* 需要注意的是不配置SMTP服务器，Jenkins也能进行test successful。这其实是说Jenkins完成了这个操作，但是操作的结果并不一定是准确的。
* 如下图是正确的配置截图和收到的邮件。

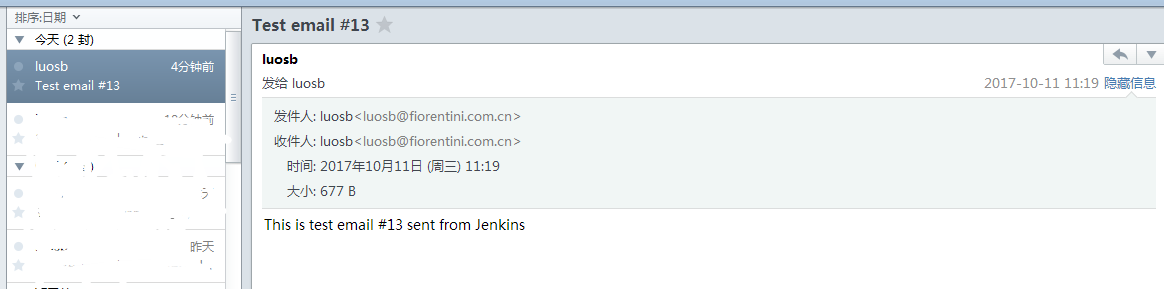
其中[SMTP 服务器]stmp.fiorentini.com.cn是公司的邮件服务器地址

[用户名]是邮件的发送者[luosb@fiorentini.com.cn](mailto:luosb@fiorentini.com.cn)，也就是luosb这个用户，下一栏的密码也要与之匹配。

[Test e-mail recipient] 是邮件的收件人



luosb收到的邮件的内容如下：



### Jenkins配置——添加slave

##### 从jenkins-master进入节点添加界面



##### 设置slave节点

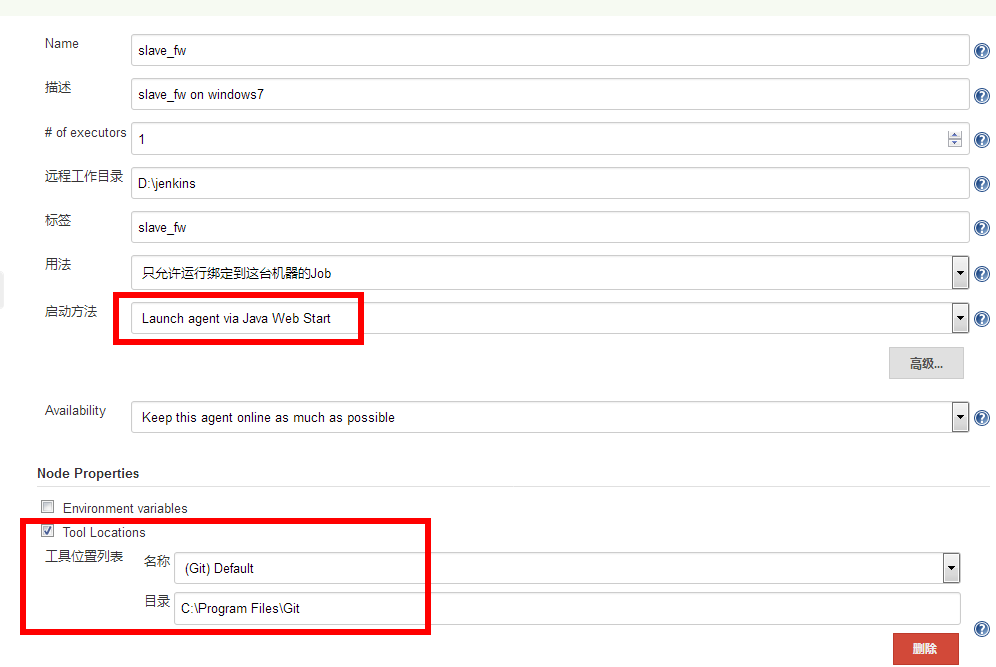
其中启动方法这里要说明：

* 选择lauch agent via java web start 是因为我们的slave是windows 7 .如果用[as windows serverice] 的方式会出现不稳定，缺少执行权限等问题 。
* Tool Locations这一项的意思是指定slave 机器需要的工具和所在路径 。一般是不用填写的。

从图中可以看出slave 需要用到git ,然后git的安装路径是C:\Program Files\Git,如果不确定的话不要乱写。

* 其他需要填写的信息

[Name]



以下是完整的参考截图和html本地文件(其中html格式的小部分与需要与服务器通信的内容会被裁掉，无法显示完整。)

* 假如你没看到“启动方法”Launch agent via Java Web Start选项,那你需要开启jnlp 端口 。
* 参考开启jnlp端口的截图 。

在jenkins的系统设置中找到[Configure Global Security]，启用安全选项.



参考解决办法 <http://www.bubuko.com/infodetail-1878706.html>

##### 在客户机启动jenkins-slave节点.

客户机上面需要安装jdk1.8共享目录中有下载[\\192.168.80.247\public\sf-auto](file:///\\192.168.80.247\public\sf-auto)。

启动slave这个节点需要在slave机器上来执行。非常重要，

启动slave这个节点采用火狐浏览器来即启动，会非常简单。

* 我们采用火狐浏览器打开jenkins-master的页面，进入slave配置项。如下图所示



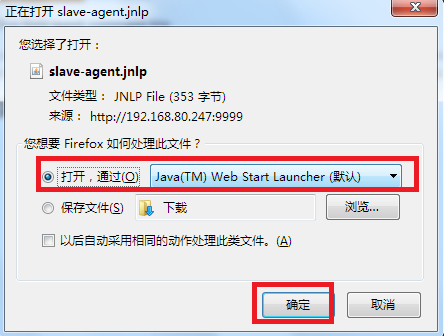
Jenkin-[系统管理]—[节点管理]--nodes配置项里面可以看到节点信息。

* 检查jenkins-slave的状态是怎么样的？

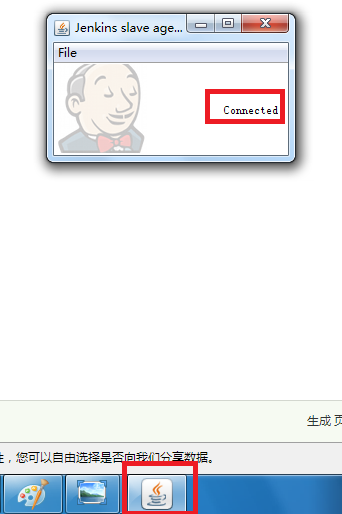
从图中红色的x可以看到slave是断开连接的。



* 我们需要用[Launch]按钮来启动slave
* 点击之后，可以看到弹出来的框 框，如下。我们选这直接打开，再确定 。



1. 接下来就要等2---5分钟，等java vm 把slave所需要的资源都在后台启动成功。
2. 启动成功后如下图。



图片中的connected表示jenkins-slave与jenkins-master已经连接上，能够进行交互了。

还会在底部保留的java图标

当你有更多的需求的时候，再考虑其他的方法来启动和制作。比如你需要自动启动jenkins-slave的时候，你需要用命令模式java –jar 的方法来启动slave。

### Jenkins配置——添加slave--实战gsmp\_fw\_slave

这个工程gsmp\_fw\_slave是在jenkins-slave机器上进行git操作，执行软件自动化测试脚本的工程实例，供大家进行slave操作的参考。

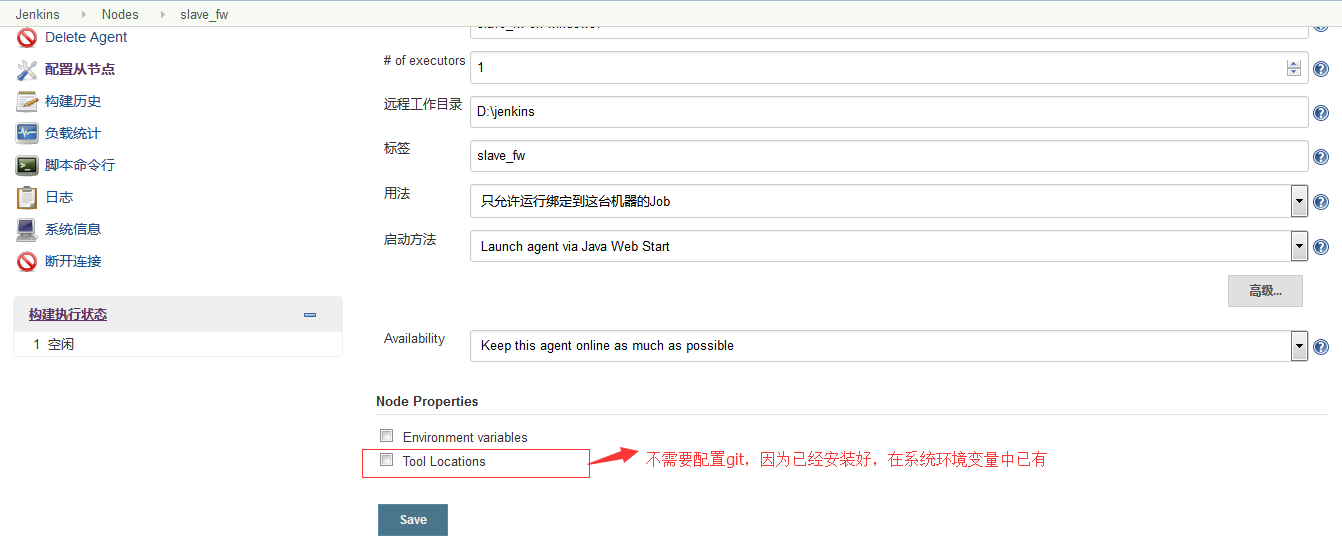
这个工程的主要作用是从gsmp-fw仓库中拿到源代码,在slave的机器上执行echo “test ok ”命令，执行ipconfig命令。

##### 检查jenkins-slave是否连接上

在jenkins主页面-系统管理 –管理节点，找到slave节点。查看状态是正常。[假如有个红色x代表连接错误]



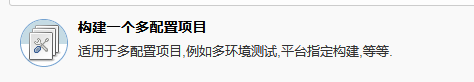
这里我们用到的git 已经安装在windows中，所有此次node节点，我们不用配置git.



Jenkins的slave节点配置图

##### 检查Jenkins-master的项目配置。

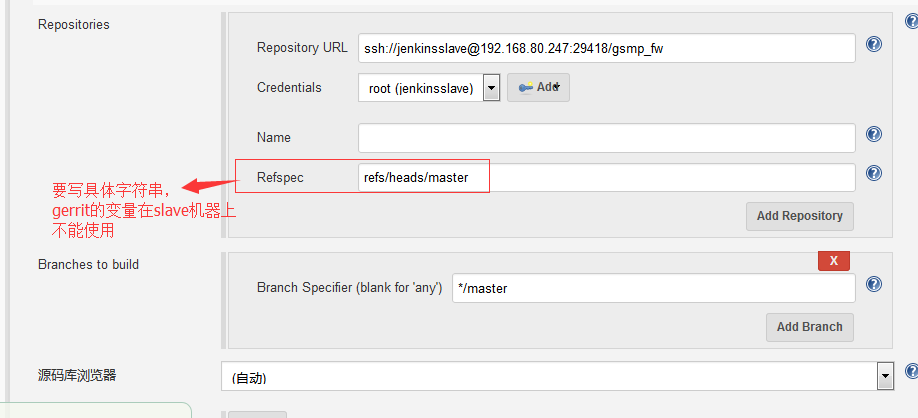
* 新建的工程需要 选择[构建一个多配置项目]，因为这里 有针对 slave的 配置项。



Jenkins的gsmp\_fw\_slave工程配置

* Gerrit的变量不能在该工程中使用

为什么呢？因为你忘了,gerrit 是安装在 Jenkins-master机器上,根本就不和slave一个机器。所以gerrit在jenkins-master机器上的环境变量在我们这个工程中是用不了的。



配置简单的命令来输出，以便我们在console中校验结果。



##### 运行起来

* 第一个灰色的gsmp\_fw是因为已经被禁止运行了。但是继续保留，供日后再启用和参考 。
* 点击构建，如图中红色框框标记的。



##### 检测结果是否正确。

通过上面已经配置的三条命令，我们可以得出正确的运行结果。

如下图：

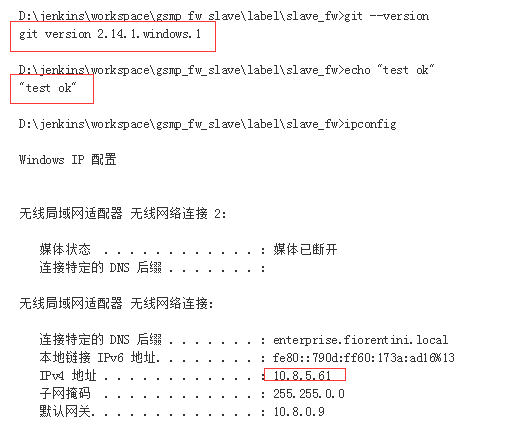
* git version 2.14.1.windows.1 ---正确
* "test ok" ---正确
* ---正确

Windows IP 配置

无线局域网适配器 无线网络连接 2:

媒体状态 . . . . . . . . . . . . : 媒体已断开

连接特定的 DNS 后缀 . . . . . . . :



##### 如何查看运行记录/log？

由于Jenkins-master 和 Jenkins-slave都需要活动，所以看Log的时候需要从jenkins-master的日志中进入Jenkins-slave的日志。



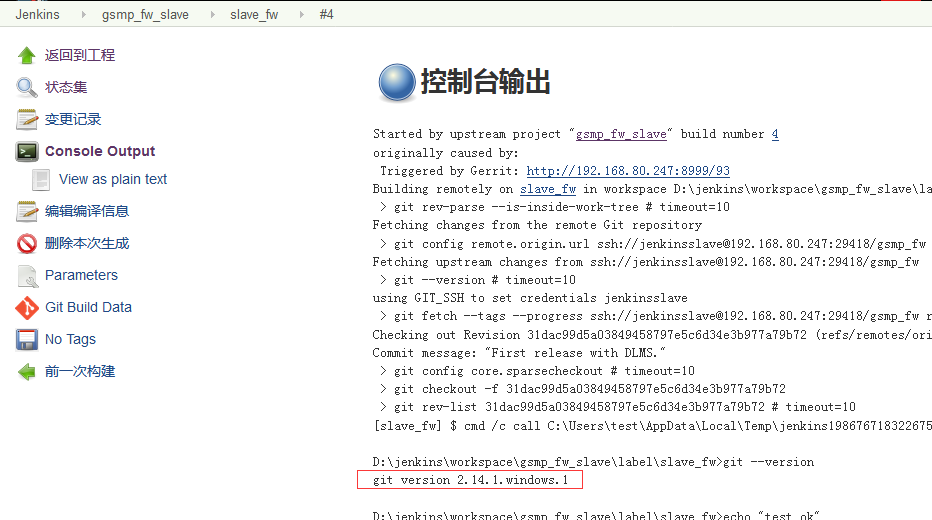
这个defualt就是入口，不要忘记，十分重要，现在执行命令是在Jenkins-slave 机器上，不是Jenkins-master机器上。



点击[console output]中可以看到运行的输出记录。



查看到 的运行记录如下图。

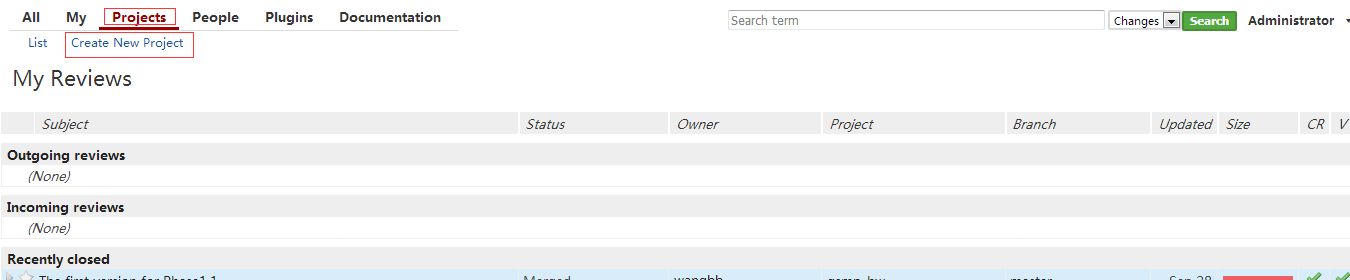


### gerrit配置---创建项目

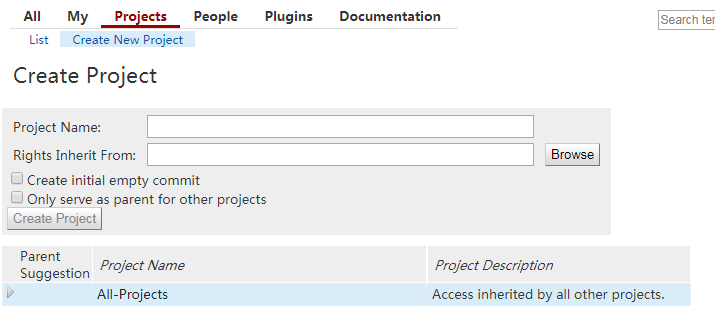
##### 使用有权限的用户登录，如Administrator

##### 进入projects菜单栏

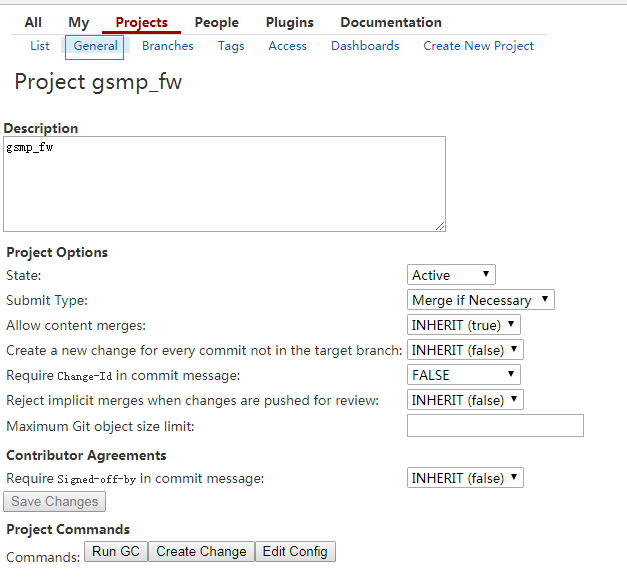
##### 点击Create New Projects



##### 填写新建项目的名字

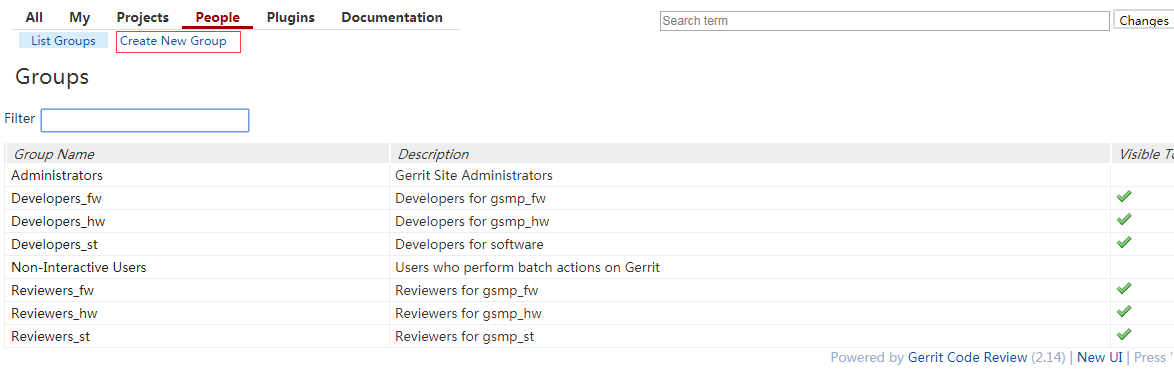


##### 配置项目的通用属性



### gerrit配置---分组

用管理员帐号Administrator，进入People 菜单栏就可以进行创建新的分组了.



Gerrit些默认的分组，如Administrators， Non-Interactive Users 等等,这些可以直接使用.

根据工程，权限的综合考虑，我们这里设定一个工程有两个组 ,他们分别是dev 组和review组。

例如硬件这个工程 gsmp\_hw，我们有Developers\_hw 组和Reviewers\_hw 组来组成。

其中Developers\_hw 组是所有组内成员，他们都可以进行git push ；Reviewers\_hw组主要是负责审核review的人员。当然负责review的也需要进行git push .也就是说Developers\_hw 组的成员是包含Reviewers\_hw 组成员的。下图中可以快速进行这项包含设置。

### gerrit配置---访问权限access

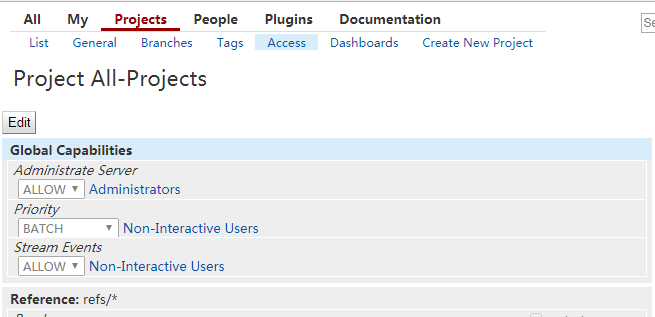
##### 项目的权限类型

第一类： 所以项目共有的权限[All-Projects]

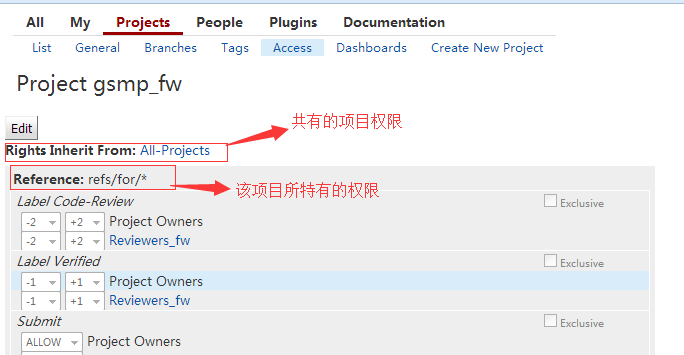
第二类： 单个项目所特有的访问权限[the project]

我们平时在设置项目权限的时候，要尽量在单个项目上设置权限。否则对日后的维护和拓展非常不利。

下图是[All-Projects] 权限简要截图



下图是[The Project] 权限简要截图



##### 项目的权限动作

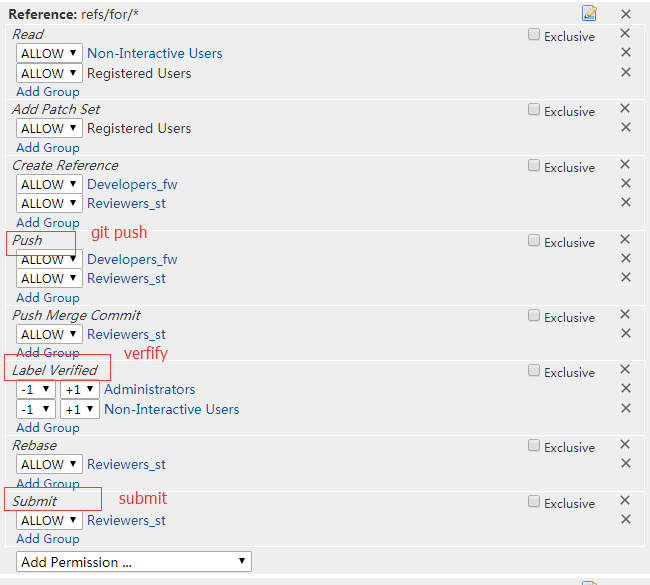
* 每个动作对应gerrit一个操作，比如

[Push ]是允许开发者 进行git push操作

[Label Verified] 是 Jenkins 用户运行脚本进行Verified 。

[Submit] 是审核者进行入库submit提交操作。

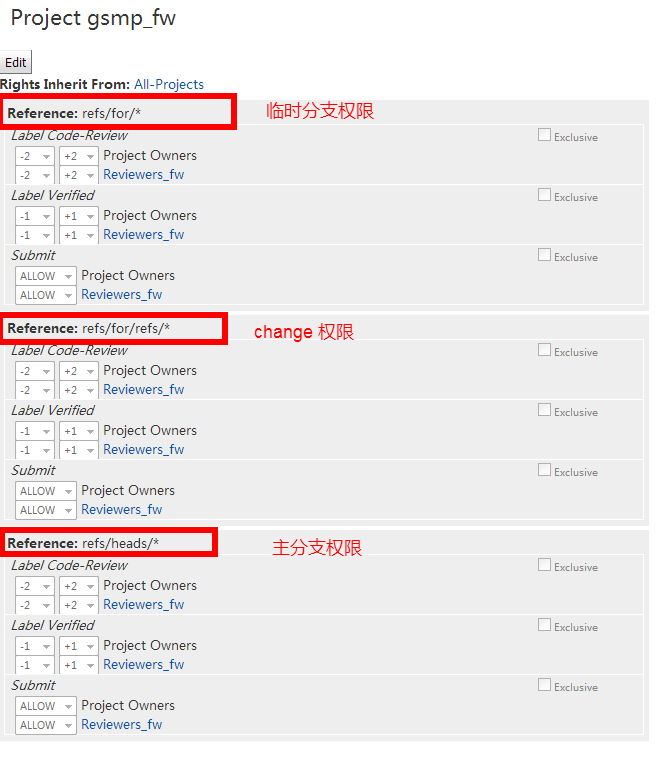
* Gerrit 的权限都是以组为单位来进行控制的，不能以单个人来进行 。如下图表示，Push 权限 是允许 两个组进行操作 Developers\_fw 和 Reviewers\_st 。



* 更多参考说明

<http://blog.csdn.net/chenjh213/article/details/50571190>

##### 项目的权限设置



这里主要有三个Reference，这里做下说明

* **refs/for/\*** 这个是临时分支需要的权限，git push过来的更新代码都是先到此目录
* **refs/for/refs/\*** 这个是changes 需要的权限,每一个小小的改动,gerrit都会有change 进行记录。
* **refs/heads/\*** 这个是主分支需要的权限 .在gerrit 里进行submit 都是把文件合并到这个分支下。

可能你还没明白为什么要有这几个分支权限，会觉得这太麻烦了。其实不是你想的那样。那是你需要理解gerrit的review机制。这里可以参考<https://zhuanlan.zhihu.com/p/21482554>

##### 项目的权限设置 参考截图和html文件

(其中html格式的小部分与需要与服务器通信的内容会被裁掉，无法显示完整。)

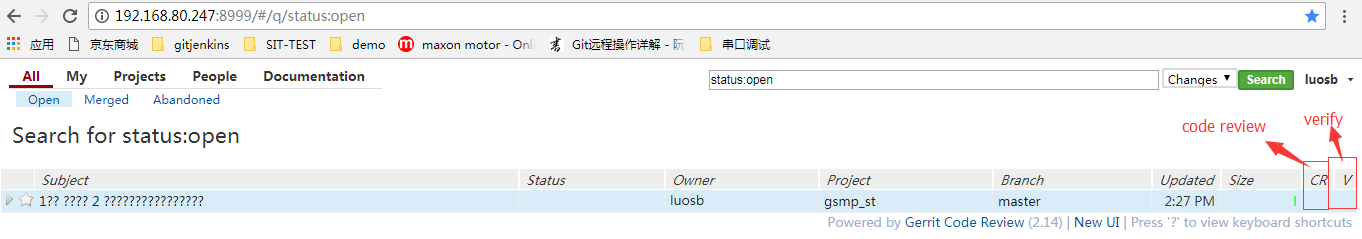
### gerrit配置---[verify][code review]

gerrit 的 verify ，可以人工进行操作核实。也可以用Jenkins来自动核实。

我们这里选择jenkins来自动verfiy 。那么怎么来实现jenkins的自动verify呢？

其实前面已经做了相关的配置。如果不记得了，还可以参考[配置jenkins中gerrit trigger](#_配置jenkins中gerrit_trigger)。

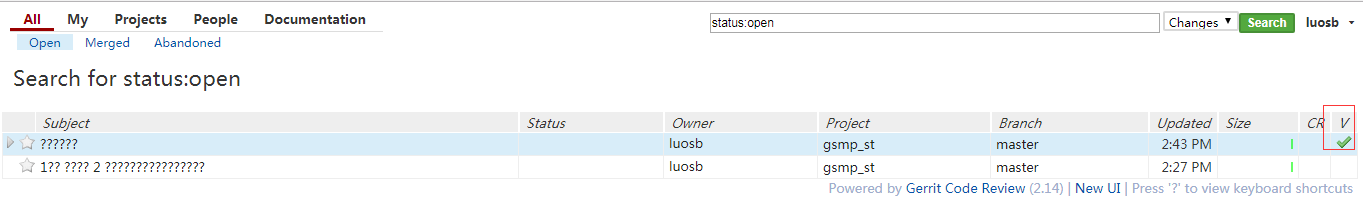
* 如下图,这是没有采用jenkins进行自动verify的截图。这时候需要有权限的用户登录并进行Verify。这并不是一个好办法。仅作演示，请继续往下看。



* 如下图,这是有采用jenkins进行自动verify的截图。

如图红色框内所示,这个更新已经被jenkins进行Verify，绿色的勾勾就是很好的标记。

其中的乱码是中文编码在gerrit未显示正常.



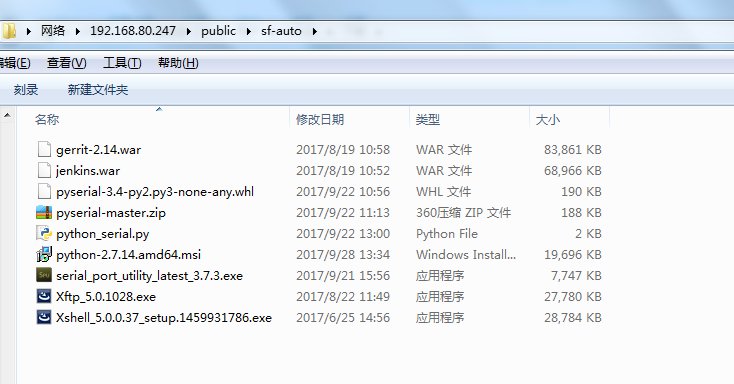
## hardware-auto-testing

### 用例图

### 环境资源使用信息

这些过程中需要用到调试工具和脚本。笔者都已保存到public目录中.

[\\192.168.80.247\public\sf-auto](file:///\\192.168.80.247\public\sf-auto)



* serial\_port\_utility\_latest\_3.7.3.exe友善串口调试助手
* python\_serial.py是Python 脚本 可以控制启动和停止电机，运行时间默认是10s。

需要安装serial模块。

关于该模块的方法是用，可以参考<http://blog.csdn.net/huayucong/article/details/48729907>

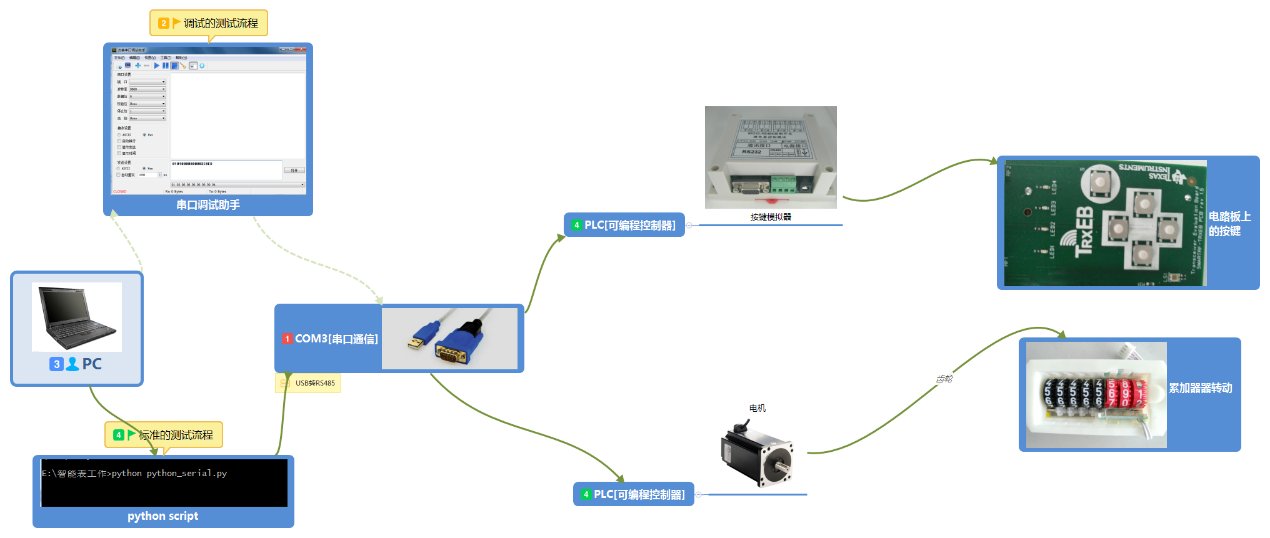
* pyserial-master.zip 是一个serial公共模块。这个采用是python的通用安装模式.可以参考命令 [python setup.py install]，如果还有不理解，可以继续参考。

注意 ：当前版本的serial采用pip安装，会遇到脚本无法饮用该模块的问题。这里推荐使用通用安装模式python setup.py install.

* python-2.7.14.amd64.msi这是Python的安装包。用过Python的童鞋都知道这是解析python脚本让计算机识别Python指令的必要环境。

### 原理图

这里仅仅是从测试开发工程师的角度来进行调试和控制。真正在项目实践中，还会有机械工程师提供的制具来包装这些工具。目前这部分的工作只是开始，还有待今后更多的整理和改进。



图中的数字说明如下,

[1 ]表示串口通信口，我们实际使用中，采用的是USB转rs485的串口通信方式，或者USB转rs232的串口通信方式。

[2] [调试的测试流程]这是在脚本不确定的情况下需要采用的调试方法—[友善串口调试助手]。该工具的安装包可以从这里下载[\\192.168.80.247\public\sf-auto](file:///\\192.168.80.247\public\sf-auto) 。有没有看到，我们可以设置左上角串口的参数信息 ，左下角可以选择命令发送的格式是十六进制，还是十进制。

而真实的命令就写在底部的框框内。接下来就可以点击打开串口和发送命令了。

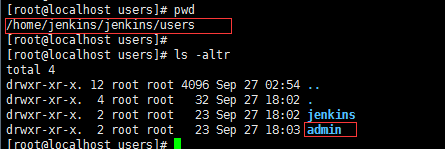
[3] [PC]表示测试使用的个人电脑。该电脑有usb接口来进行通信。

[4] [标准的测试流程]真实项目的测试流程应该是这样的。调用脚本来控制电机或者其他支持PLC的设备。图中设备包括案件模拟器和步进电机。

## FAQs常见问题解答

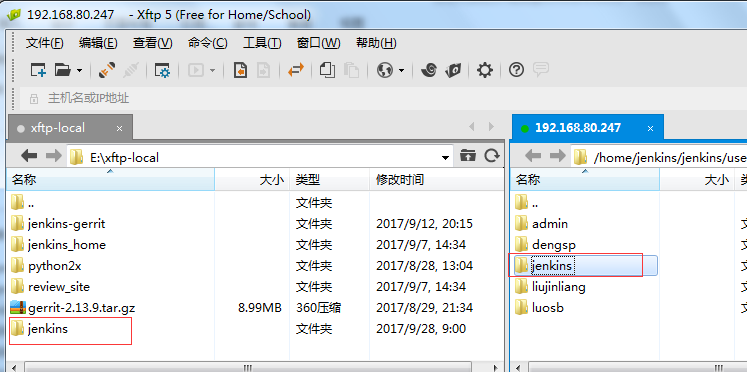
### Jenkins的admin用户不能登陆了，怎么办？

出现这个问题，不要着急，这里有个很好方法，可以迅速解决，那就是修改Ｊｅｎｋｉｎｓ的用户配置文件．那么怎么修改呢？　请看下图



从图中，我们可以看出ｊｅｎｋｉｎｓ的用户配置文件在/home/Jenkins/Jenkins/users目录下，这里有两个用户　ａｄｍｉｎ和　ｊｅｎｋｉｎｓ。你是不是想到了办法――――替换文件。

没错就是替换文件就可以了。下图可以看到，用ｘｓｈｅｌｌ的兄弟软件　ｘｆｔｐ就轻轻松松的把文件从ｗｉｎｄｏｗｓ　系统　传输　到Ｃｅｎｔｏｓ７．１的Ｊｅｎｋｉｎｓ工作目录下　．



* 更多参考：<http://bookong.iteye.com/blog/1964182>

### Jenkins slave 没有java web start这个启动项

* 假如你没看到“启动方法”Launch agent via Java Web Start选项,那你需要开启jnlp 端口 。
* 参考开启jnlp端口的截图 。

在jenkins的系统设置中找到[Configure Global Security]，启用安全选项.



参考解决办法 <http://www.bubuko.com/infodetail-1878706.html>

### 为什么要用gerrit来进行code review ?

开发人员一个人写代码是很容易出错的，这些小小的错误，会对整个项目有不可估计的风险。比如开发人员A 优化了主要功能支付，他以为是很小的优化，对整个代码没改动。但是开发人员B 是从主要支付功能基础上做了新的功能。这个时候，就需要进行review,来预测开发人员A的优化对开发人员B是否没有影响。

**综合来说code review的意义：**

1. 首先，代码审查的最大的功用是纯社会性的。如果你在编程，而且知道将会有同事检查你的代码，你编程态度就完全不一样了。你写出的代码将更加整洁，有更好的注释和程序结构。
2. 其次，偷懒是人的天性，从节约成本的角度考虑，大家一般会选择在测试之前无限制的Delay Code Review。入库前做Code Review便是成本和效果之间最佳平衡点，它能及时发现问题，进行修改后确保代码质量。
3. 最后，代码审查能传播知识。在很多开发团队里，经常每个人负责一个核心模块，每个人都只关注自己的模块。除非是同事的模块影响了自己的程序，他们从不相互交流。这种情况的后果是，每个模块只有一个人熟悉里面的代码。如果这个人休假或辞职了，其他人则束手无策。通过代码审查，至少会有两个人熟悉这些程序——作者，以及审查者。审查者并不能像程序的作者一样对程序十分了解，但至少他会熟悉程序的设计和架构，这是极其重要的。

### 由于主机IP地址更换,安装在jenkins中的gerrit trigger插件不能触发怎么办？

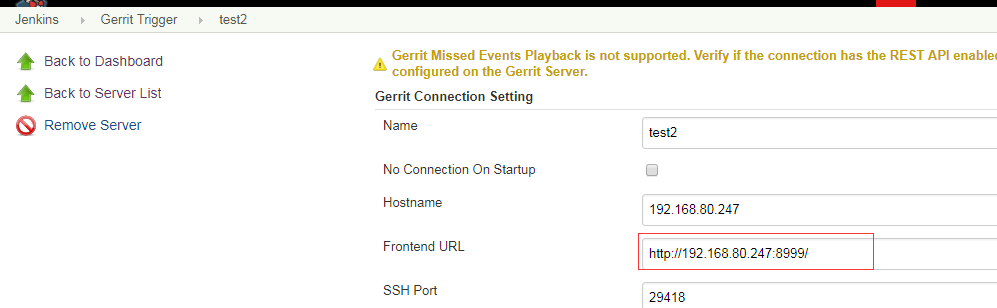
* [方案一 ]：最直接，最可靠的办法是重新部署jenkins,gerrit这一整套系统。但是这样会花费大量时间，尤其是对于没有部署过的新手来说，简直太可怕了。
* [方案二 ]: 修改所有相关的IP地址。理论上这是可行的，但是在实际调试中，笔者未能找到所有需要修改的IP地址，结果就是怎么也没有再触发gerrit。

目前已知的需要修改的地方有:

① gerrit的地址配置

打开文件后有个”canonicalWebUrl = [http://192.168.80.247:8999”其中192.168.80.247](http://192.168.80.247:8999)就是需要修改的

② jenkins中修改gerrit trigger下的FrontendUrl.



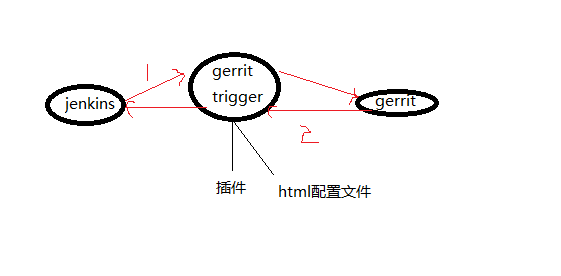
* [方案三]：删除jenkins的gerrit trigger插件，再重装.

① 卸载插件gerrit trigger

② 删除jenkins下关于gerrit的配置文件

③ 再安装gerrit trigger

这里再重复一下jenkins是通过成千上万的插件（plugins）来进行集成，才能集成成千上万的好工具的，比如git ,gerrit,robotframework。对于我们这里的gerrit来说，是通过gerrit trigger这个插件来与gerrit进行集成的。



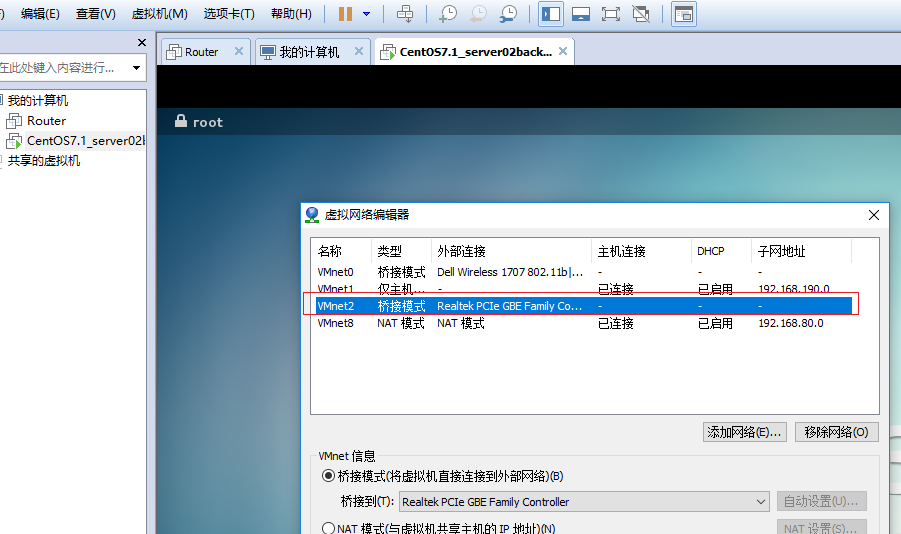
* 最后，说明一下想办法恢复IP地址是更好的方法，其次才是重新安装插件和gerrit 。

### 虚拟机在vmware中进行还原后，IP不一致怎么办？

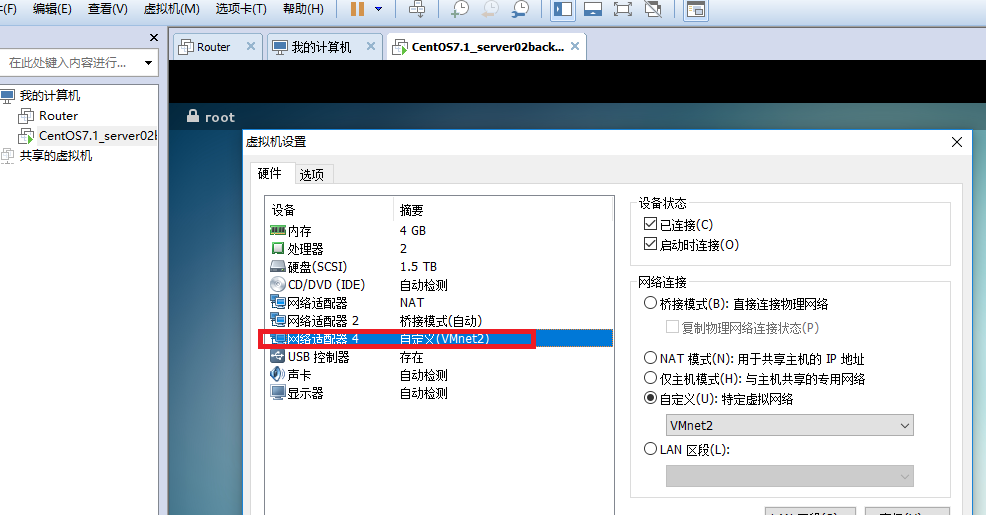
最好的办法把新centos7.1这个虚拟机的IP地址设定为和原来一样.比如都是192.168.80.247，再关闭老的虚拟机。

这里介绍vmware新增桥接的方法。

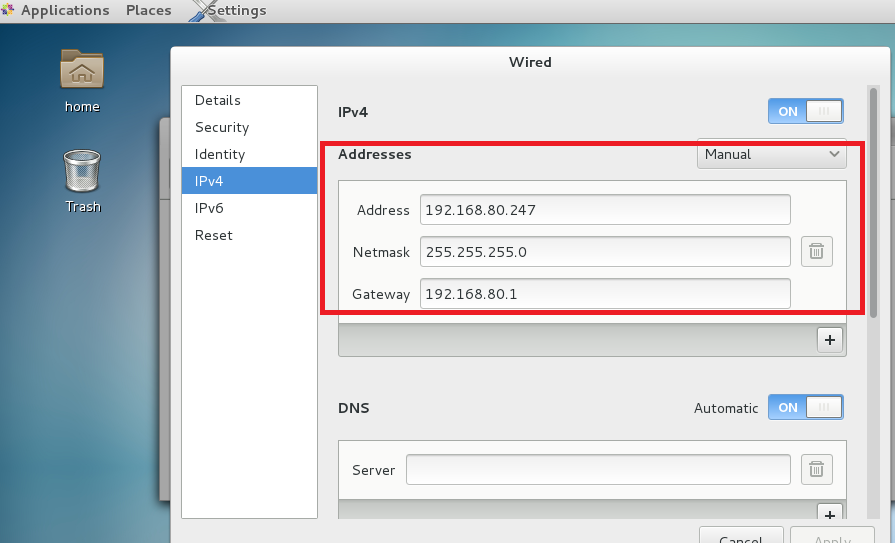
1. 在vmware中增加编辑虚拟机网络并增加桥接网卡vmnet2



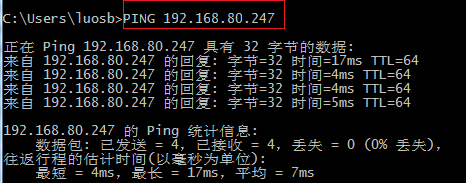
1. 在vmware中选择该虚拟机，并且进行网络设置。把之前增加的网络vmnet2添加到该虚拟机中。



3 在 虚拟机中配置IPV4的固定地址。下图是图形化的编辑，当然你也可以采用编辑文档的方法，比如 vim /etc/sysconfig/network-scripts/ifcfg-ens33 .



3 在其他电脑上用Ping来检测网络是否OK。如下图已接收到4个数据包。



### 这个文档太长，太难掌握，找不到想要的答案，怎么办？

其实这个不难，亲爱的读者请你静下心来慢慢看。其实这里有个很好的目录，可以帮你快速了解文档的结构，还有可以伸缩的导航帮你快速定位文档位置。

