

Jenkins安装文档

1 Docker下安装Jenkins

本教程在docker下安装Jenkins：

拉取镜像：

docker pull jenkinsci/blueocean

创建容器：

```
docker create --name jenkins -u root -p 8889:8080 --privileged=true -v jenkins-  
data:/var/jenkins_home -v /var/run/docker.sock:/var/run/docker.sock -v /home/jenkins:/home  
docker.io/jenkinsci/blueocean
```

启动容器：

docker start Jenkins

输入<http://192.168.101.64:8889>

初次运行Jenkins会经过一个安装过程，一般情况使用默认配置，下一步安装即可，其中一步需要输入密码，如下图：



出现上边的画面需要进入Jenkins容器查看密码，并将密码拷贝到上图输入框中：

```
docker exec -it jenkins /bin/bash
cat /var/jenkins_home/secrets/initialadminPassword
```

2 Jenkins全局工具配置

本项目使用Jenkins需要配置Jdk1.8、Git、maven。

1、Maven安装：



可以采用“自动安装”，在上图中勾选自动安装，选择maven版本即可。

本教程将maven程序目录拷贝到宿主机/home下，在上图配置即可。

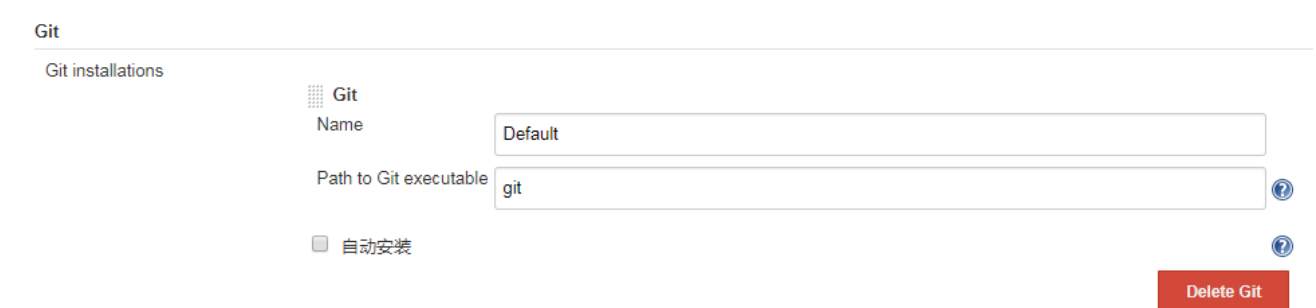
2、Jdk安装配置



与maven安装相同，也可以采用自动安装或手动安装的方法。

3、Git安装方法同上

git安装完成配置如下图：



3 SSH Remote Hosts

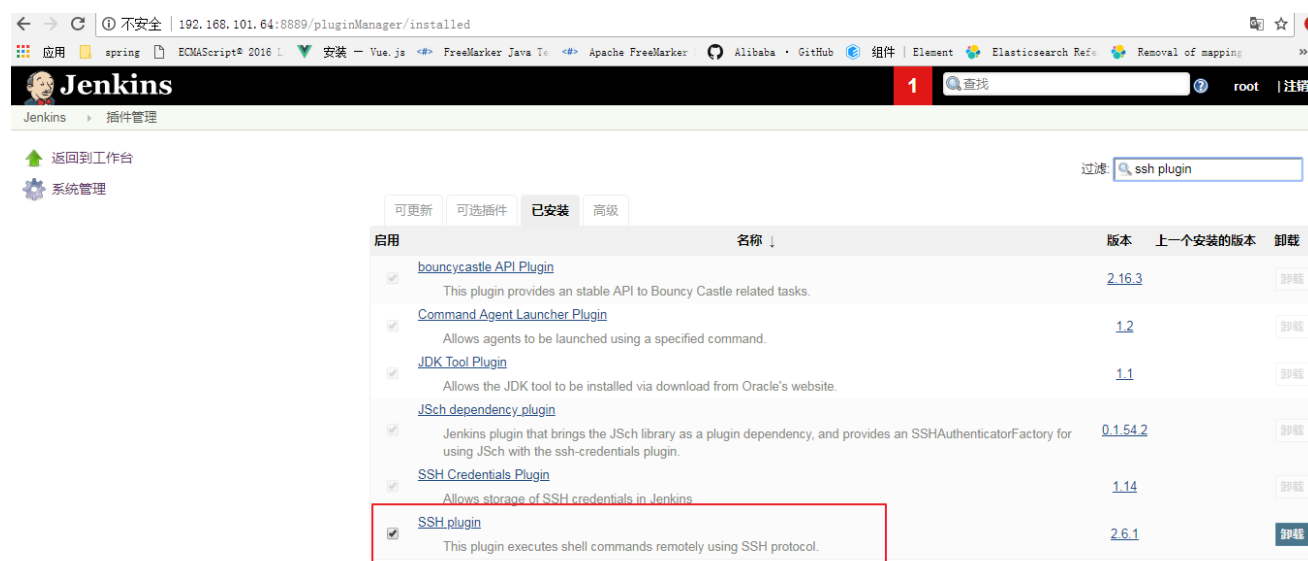
SSH remote hosts是Jenkins提供的一种远程访问ssh服务器的方法，通过如下步骤测试此功能：

1、安装插件SSH plugin

从可选插件中选择SSH Plugin进行安装

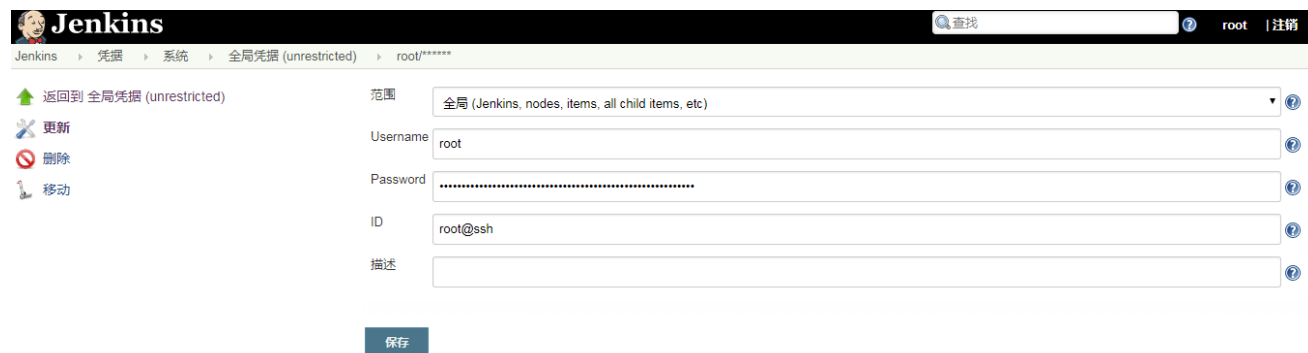


安装成功可在已安装插件中查询：



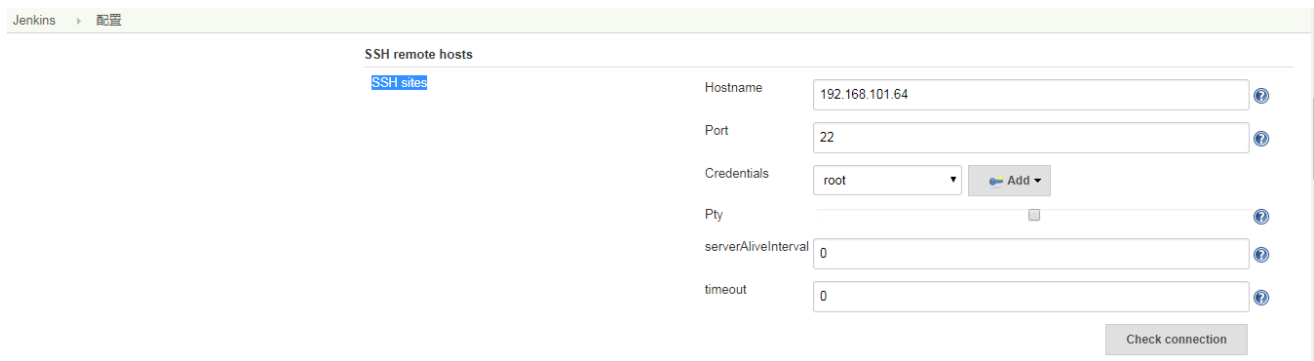
2、配置凭证

配置远程SSH服务的账号和密码：



3、配置SSH sites

配置远程ssh服务器的ip和端口，并选择第2步设置凭证



Jenkins > 配置

SSH remote hosts

SSH sites

Hostname: 192.168.101.64

Port: 22

Credentials: root

Pty: ☐

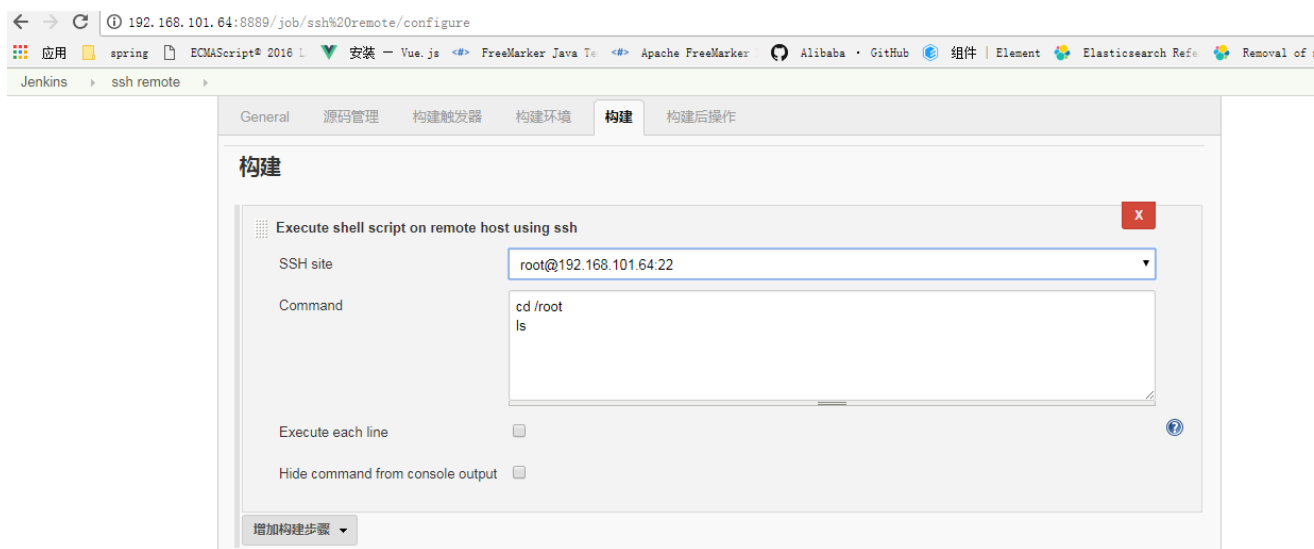
serverAliveInterval: 0

timeout: 0

Check connection

4、在任务构建中编写脚本

选择第3步定义的SSH site，并在command输入脚本。



192.168.101.64:8889/job/ssh%20remote/configure

应用 spring ECMA Script 2016 L 安装 - Vue.js <#> FreeMarker Java 7 <#> Apache FreeMarker Alibaba · GitHub 组件 | Element Elasticsearch Refs Removal of

Jenkins > ssh remote >

General 源码管理 构建触发器 构建环境 构建 构建后操作

构建

Execute shell script on remote host using ssh

SSH site: root@192.168.101.64:22

Command: cd /root
ls

Execute each line: ☐

Hide command from console output: ☐

增加构建步骤

配置完成可以测试是否可以远程执行SSH指令。

4 搭建Docker私有仓库

微服务的镜像会上传到Docker仓库保存，常用的公网Docker仓库有阿里云，网易云等，在企业局域网也可以搭建自己的Docker私有仓库，本教程使用Docker提供的私有仓库 registry。

进入Docker私有仓库所在服务器执行：

1、安装Docker

2、创建私有仓库容器

```
docker run --name docker-registry -d -p 5000:5000 registry
```

执行结果如下：

```
[root@localhost xc-govern-center]# docker run --name docker-registry -d -p 5000:5000 registry
Unable to find image 'registry:latest' locally
Trying to pull repository docker.io/library/registry ...
latest: Pulling from docker.io/library/registry
4064ffdc82fe: Pull complete
c12c92d1c5a2: Pull complete
4fbc9b6835cc: Pull complete
765973b0f65f: Pull complete
3968771a7c3a: Pull complete
Digest: sha256:51bb55f23ef7e25ac9b8313b139a8dd45baa832943c8ad8f7da2ddad6355b3c8
Status: Downloaded newer image for docker.io/registry:latest
aaf774e6dcc76173dceb4ff16831a129238e7dbcd770d48c25e214b972c9553
[root@localhost xc-govern-center]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED
aaf774e6dcc7	registry	"/entrypoint.sh /e..."	20 seconds ago
Up 20 seconds	0.0.0.0:5000->5000/tcp	docker-registry	

上边显示 registry 镜像已经创建，并且 docker-registry 容器已经启动成功。

访问：http://192.168.101.64:5000/v2/_catalog

响应结果如下：

```
{"repositories":[]}
```

上边的响应结果说明在 docker 私有仓库中还没有镜像。

默认 docker-registry 只允许 https 提交镜像，如下配置使 docker-registry 支持 http

在 /etc/docker 下，创建 daemon.json 文件，写入：

```
{ "insecure-registries":["192.168.101.64:5000"]}
```

重启 docker：

```
systemctl restart docker.service
```

启动 docker-registry

```
docker start docker-registry
```

