Docker镜像生成和部署

本文基于项目实战，描述如何通过docker镜像部署项目，本文所用的工具如下。

IDE： STS，安装docker tooling插件

Docker仓库：使用的阿里云公有docker仓库，地址：<https://dev.aliyun.com>

Docker daemon：运行在centos 7环境下

本机docker环境：docker toolbox

# 服务端docker daemon配置

Docker tooling 连接docker daemon有两种方式，一种是连接非加密通讯的docker daemon，另一种是连接加密通讯的docker daemon。

由于本机使用的是docker toolbox，默认使用了加密方式通讯，以下只介绍加密通讯时的配置。非加密方式不介绍，也比较简单。

## 远程主机docker daemon配置

默认的docker daemon并不是以加密方式通讯，需要配置加密方式进行通讯。此节说明的配置需要服务器维护人员进行配置。

### 生成tls加密所需的文件

创建一个文件夹

mkdir /ssl

cd /ssl

创建ca密钥

openssl genrsa -aes256 -out ca-key.pem 4096

创建ca证书

openssl req -new -x509 -days 1000 -key ca-key.pem -sha256 -subj "/CN=\*" -out ca.pem

创建服务器私钥

openssl genrsa -out server-key.pem 4096

签名私钥

openssl req -subj "/CN=\*" -sha256 -new -key server-key.pem -out server.csr

使用ca证书与私钥证书签名

openssl x509 -req -days 1000 -sha256 -in server.csr -CA ca.pem -CAkey ca-key.pem -CAcreateserial -out server-cert.pem

生成客户端密钥

openssl genrsa -out key.pem 4096

签名客户端

openssl req -subj "/CN=client" -new -key key.pem -out client.csr

创建配置文件

echo extendedKeyUsage=clientAuth > extfile.cnf

签名证书

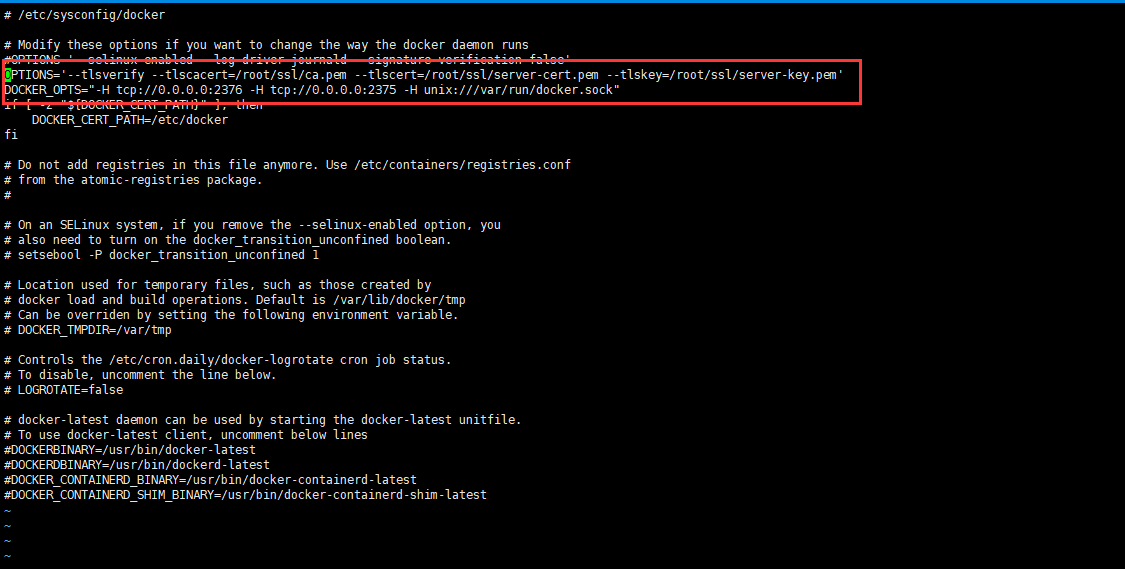
openssl x509 -req -days 1000 -sha256 -in client.csr -CA ca.pem -CAkey ca-key.pem -CAcreateserial -out cert.pem -extfile extfile.cnf

删除多余文件

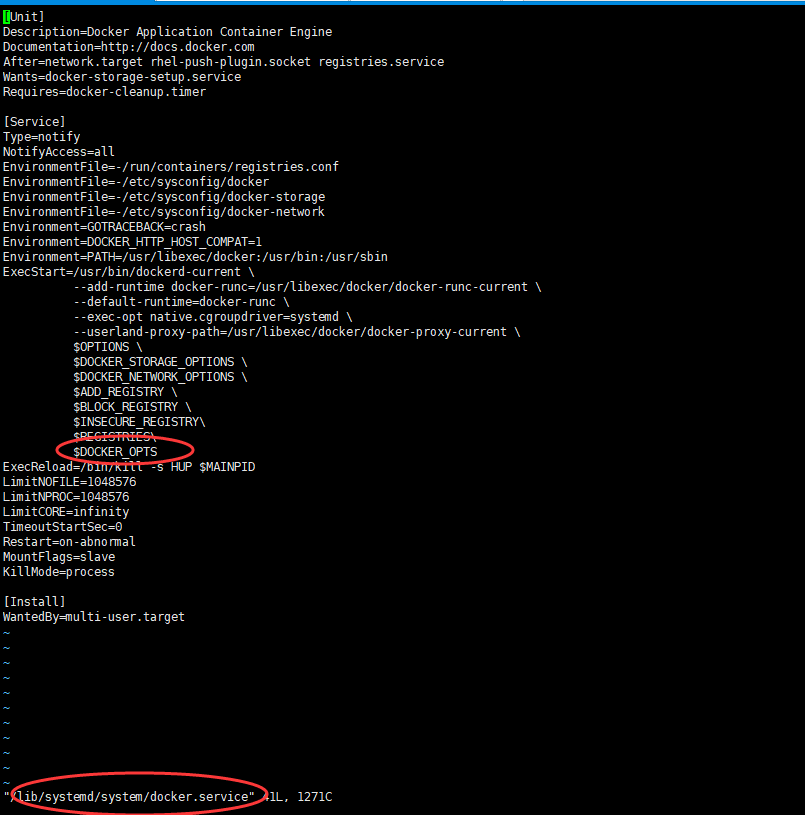
rm -rf ca.srl client.csr extfile.cnf server.csr

### 配置docker服务启动参数

修改docker启动配置文件，添加下图中红框标示的参数，加密所用的文件路径改为用户自己生成加密文件所在的路径。



修改docker服务启动服务参数，如下图



重启后台进程

systemctl daemon-reload

重启docker服务

systemctl restart docker

将 /ssl/ca.pem /ssl/cert.pem /ssl/key.pem 三个文件复制需要连接远程docker daemon的主机

# 本机环境配置

## 本机docker环境变量配置

Docker toolbox安装不在介绍，官网有详细的安装过程说明。

官网地址：<https://docs.docker.com/toolbox/toolbox_install_windows/>

检查docker命令是否正常，在命令行终端输入docker images，能正常显示说明没有问题，否则需要配置环境变量。

配置信息如下：

DOCKER\_TLS\_VERIFY=1

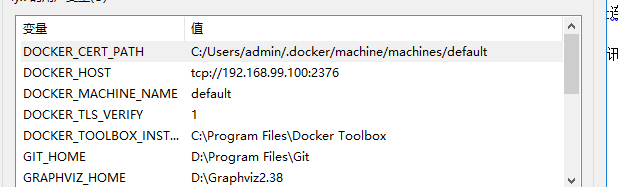
DOCKER\_HOST=tcp://192.168.99.100:2376

DOCKER\_CERT\_PATH=C:\Users\admin\.docker\machine\machines\default //路径配置成实际的路径

DOCKER\_MACHINE\_NAME=default

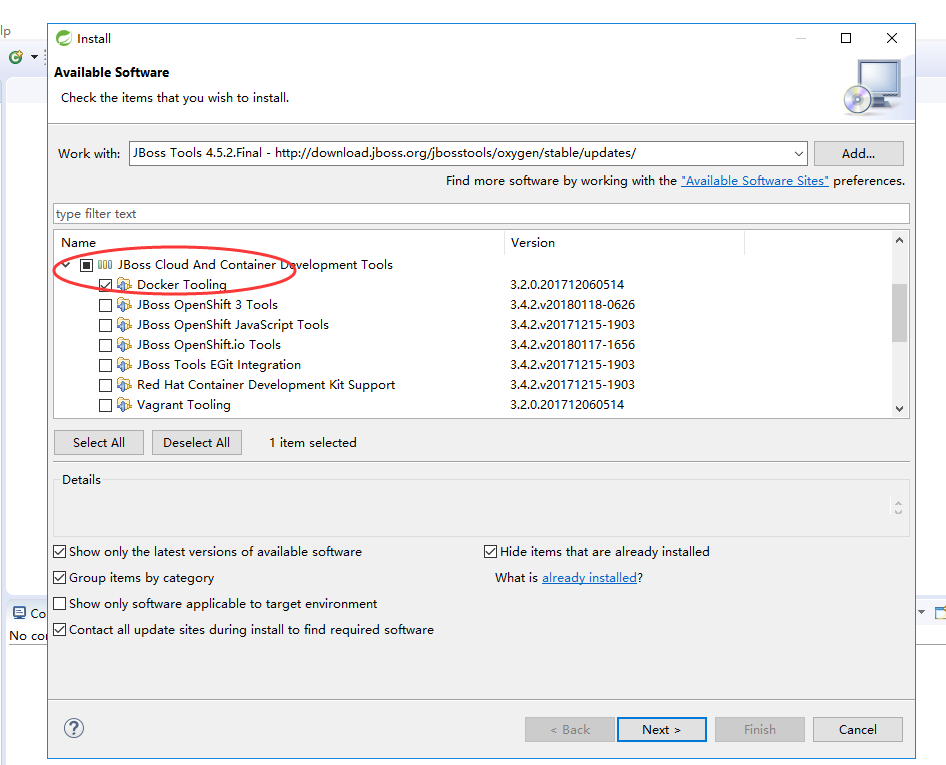
COMPOSE\_CONVERT\_WINDOWS\_PATHS=true

配置完成如下图：

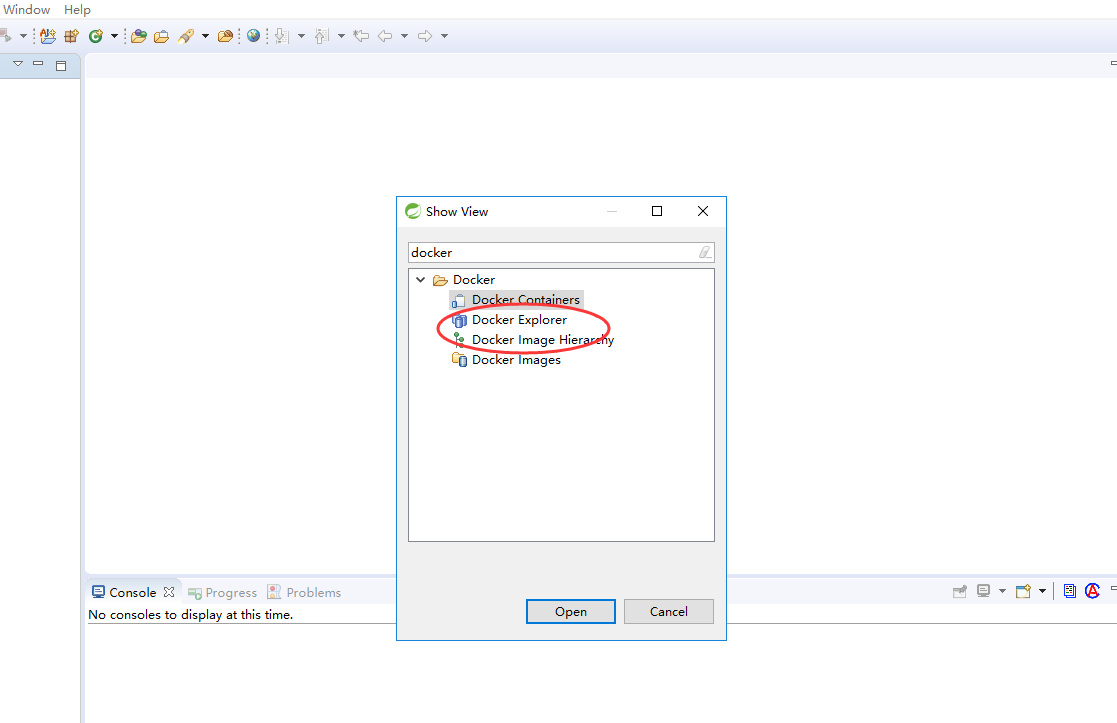


## 配置STS docker tooling插件

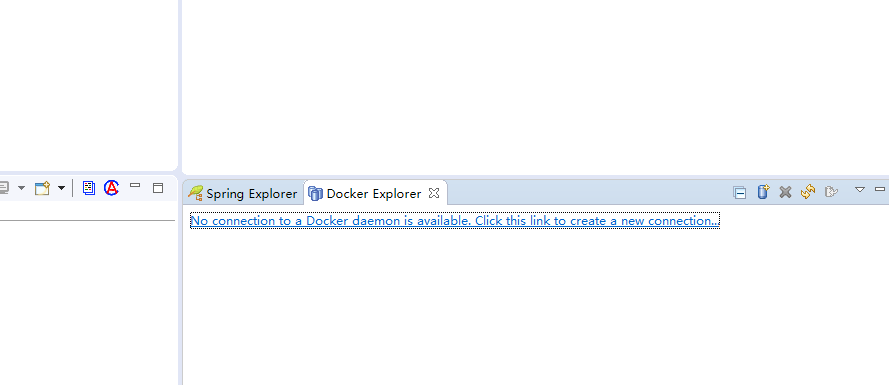
首先需要在STS上配置docker管理插件docker tooling，插件安装说明和下载地址见官网说明，官网地址：<http://tools.jboss.org/downloads>。



安装完成后，调出docker explorer 视图，如下图：

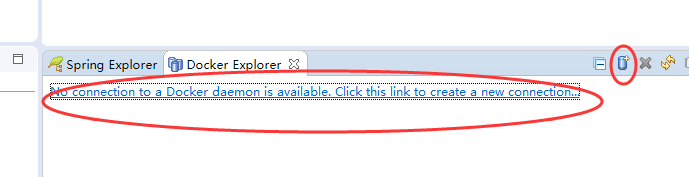


Docker explorer 视图如下：

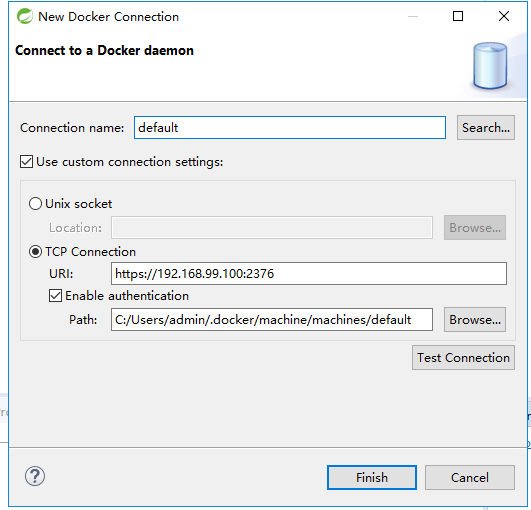


## 添加远程主机docker连接

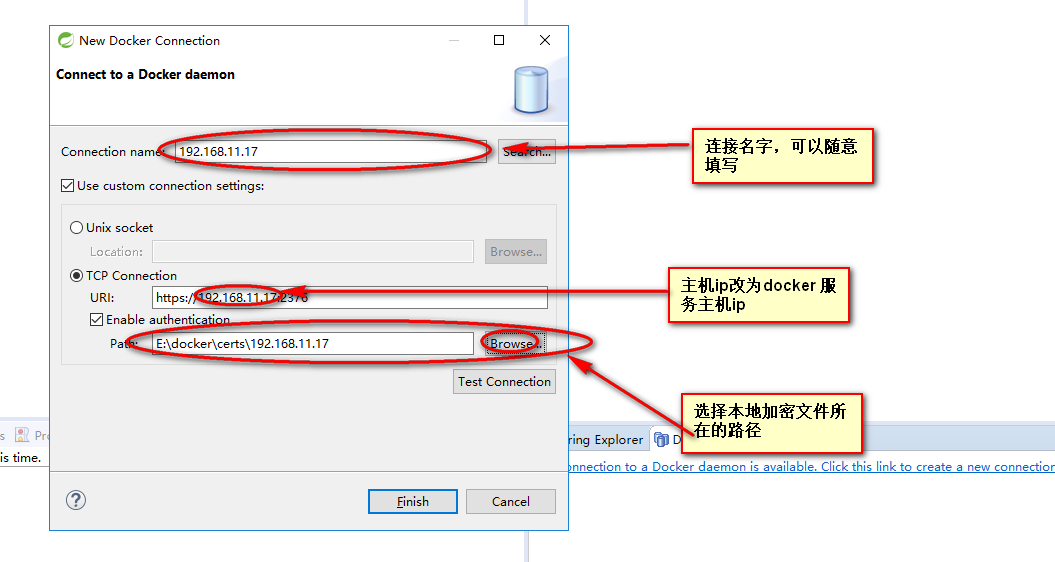
如下图，添加红圈标示的区域均可添加连接。



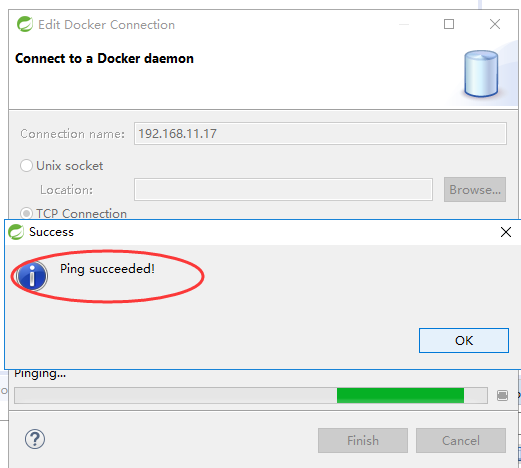
点击后，弹出如下的弹出框：



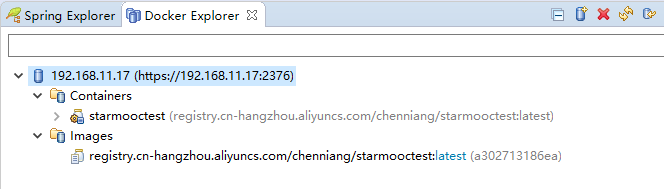
修改各个参数，连接的远程docker服务器地址为192.168.11.17，配置完成如下图：



点击test connection按钮，弹出如下图所示ping success表示可以正确连接。



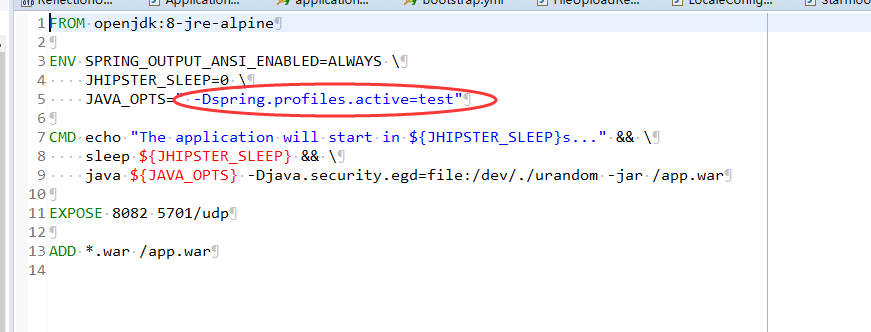
如下图，可以看到docker daemon上面存在的镜像和创建的容器。



# 生成docker 镜像

项目使用maven管理包依赖，编译打包和docker镜像生成，maven pom文件不再介绍，使用jhipster生成的项目已经自动生成pom文件。

Docker镜像生成使用jhipster 生成的dockerfile文件生成。简单修改下dockerfile文件，添加镜像启动的时激活的配置文件。如下图：



执行如下命令生成镜像：

mvn clean package -Ptest dockerfile:build –DskipTests

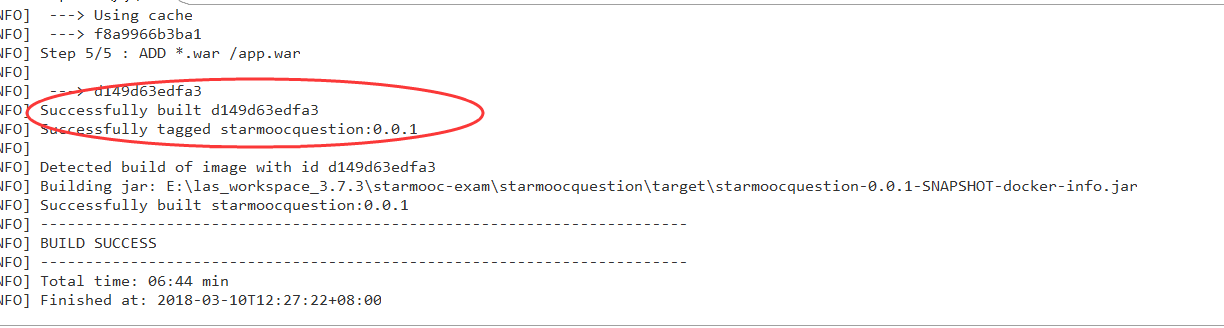
说明：

-Ptest表示使用pom.xml里面id为test的profile配置；

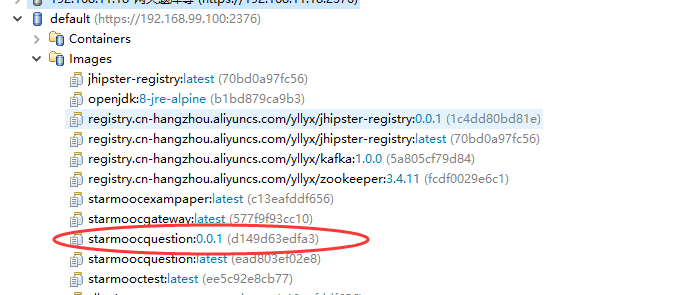
-DskipTests表示忽略所有与测试相关的编译过程；

dokerfile:build表示生成docker镜像。

如下图所示，表示生成镜像成功。



在docker explorer里面也能查看到生成镜像，如下图：

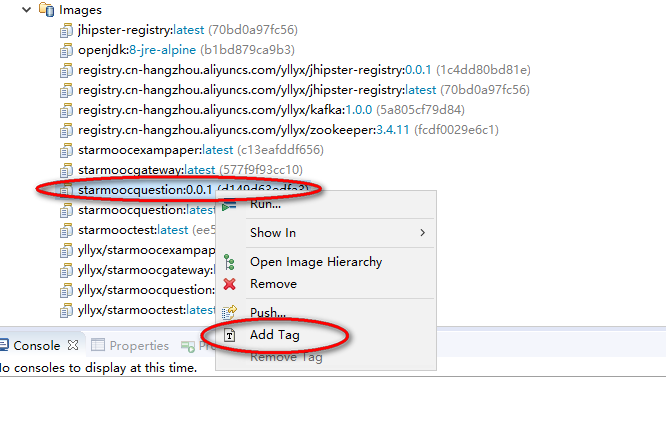


# 部署镜像

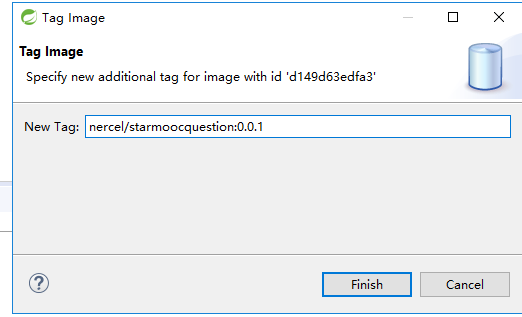
第3节生成的镜像在本地环境，部署镜像需要将镜像push到镜像仓库，镜像仓库可以是私有仓库，也可以是公有仓库。我们暂时使用的是阿里云的docker仓库。官网地址为：<https://dev.aliyun.com>。

阿里云镜像仓库创建不介绍。本文直接使用已经创建好的镜像仓库。

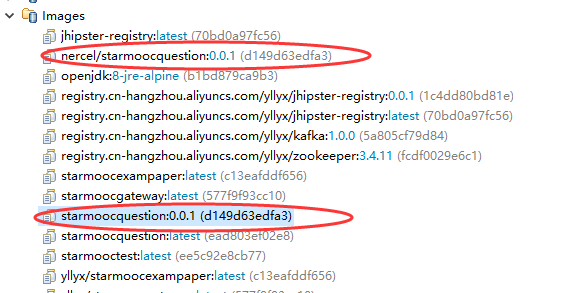
首先给生成的docker镜像打上符合阿里云要求的tag。鼠标右键点击需要添加tag的镜像，如下图，选择add tag菜单。



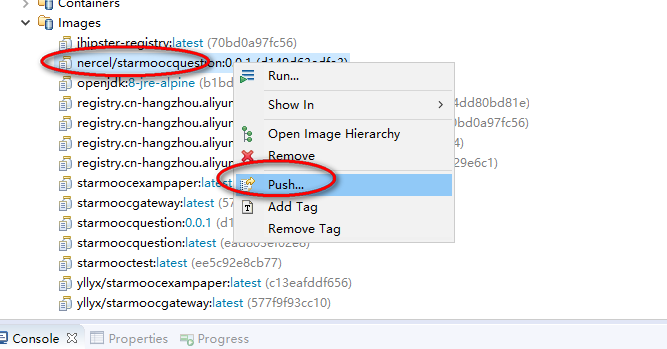
如下图，在弹出对话框中填写tag信息。



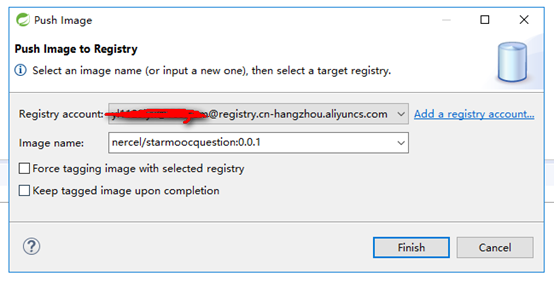
点击finish按钮，如下图，同一个镜像id,有两个tag。



如下图，鼠标右键点击镜像，选择push菜单。

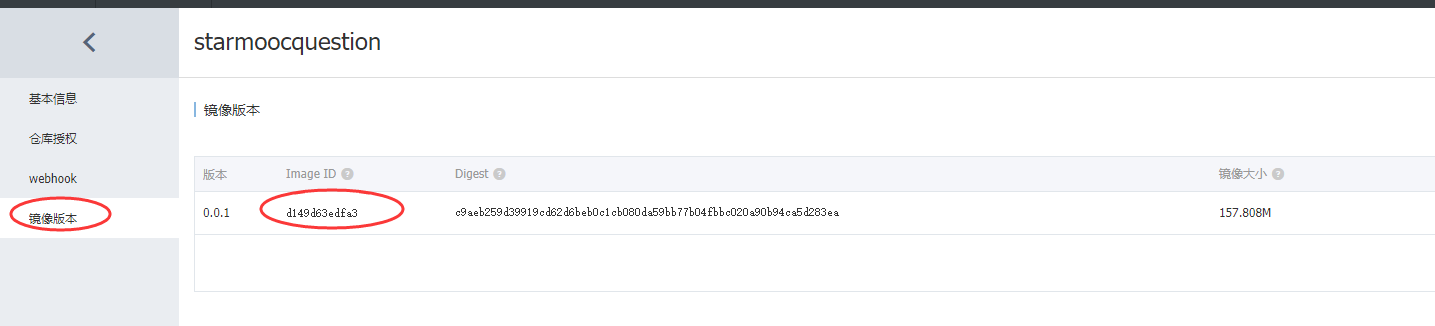


如下图，弹出如下对话框。



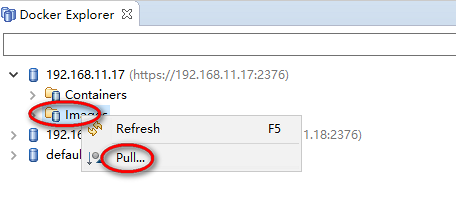
Registry account阿里云docker仓库注册的账号信息。点击finish，开始上传镜像。

镜像上传完成后，可以在阿里云仓库中参看镜像信息。如下图：

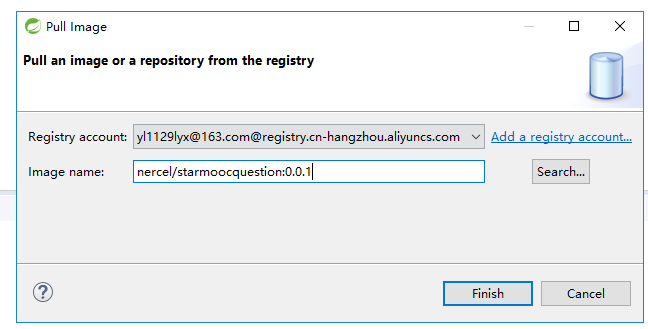


镜像上传完成后，就可以在利用此镜像进行部署。

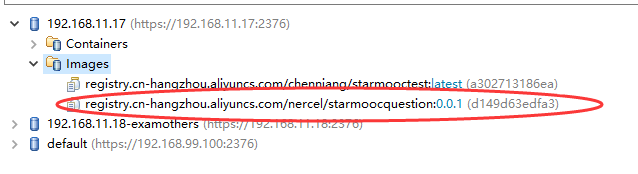
先使用STS docker tooling 连接远程docker，连接成功后，如下图，右键点击Images,选择pull。



如下图，再弹出框中填写Image name。点击finish，开始下载。



下载完成，可以查看到已下载的镜像。如下图：



利用下载的镜像部署不在介绍，docker tooling图形化界面，每个步骤均有说明。