**AWS动手实验ECR**

**实验目的**

新建一个存储库  
上传一个镜像到新建的库中

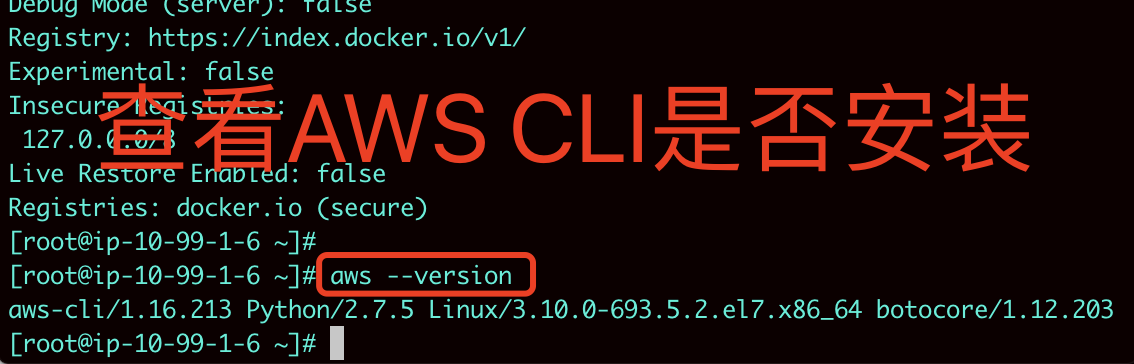
**实验要求**

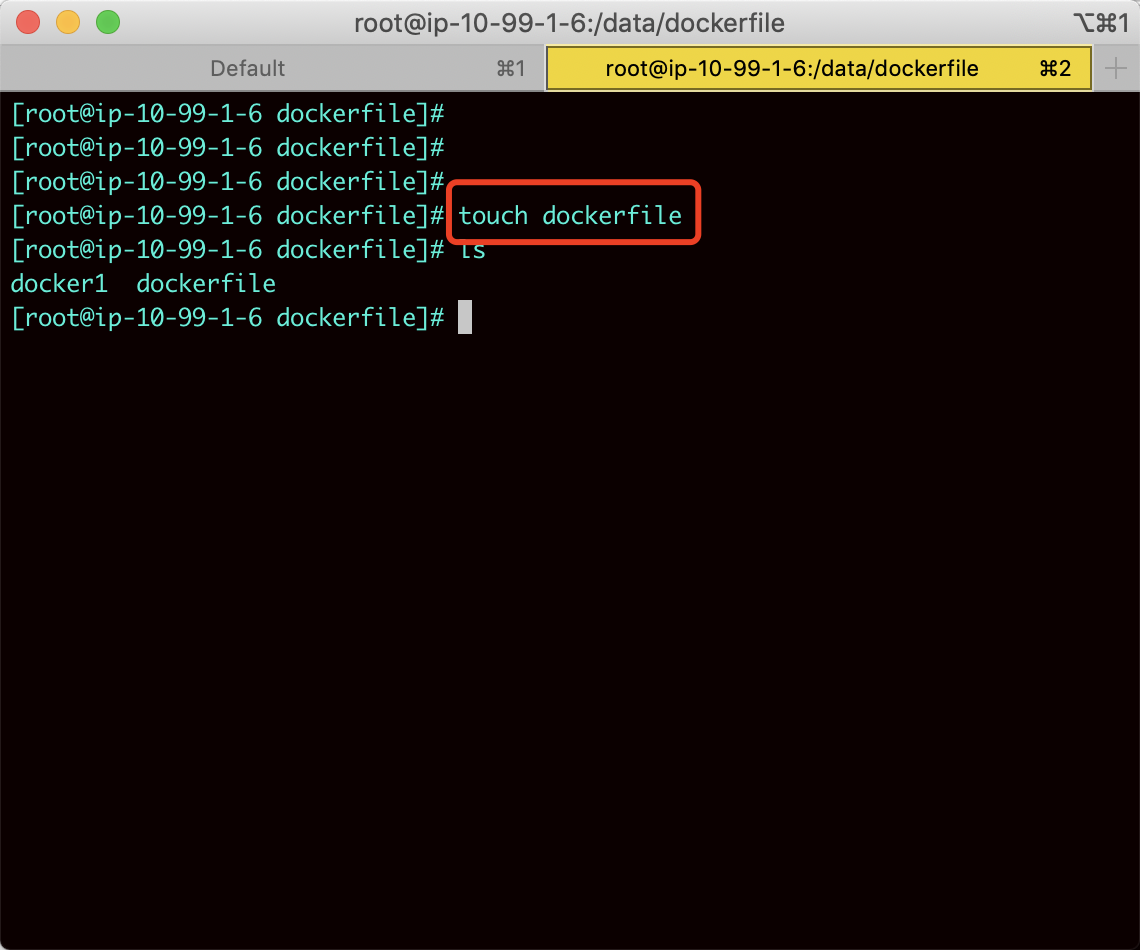
了解Amazon Web Services  
准备安装好AWS CLI与docker的Linux主机  
熟悉docker的使用  
了解Amazon Elastic Container Registry

**实验原理**

Amazon Elastic Container Registry（Amaozn ECR）是一项托管AWS Docker镜像仓库服务，开发人员可使用它轻松存储、管理和部署Docker容器镜像。您可以将Amazon ECR和Amazon Elastic Container Service（ECS）结合使用，以便在开发、测试和生产环境下实现安全轻松的协作和部署。

**实验步骤**

1、登陆到预先安装并配置好AWS CLI与docker的Linux主机，如下：  
  


2、创建一个简单的Web应用程序的Docker镜像：  
$nbpa.新建一个名称为【dockerfile】的文件，（dockerfile是一个清单，描述了要用于Docker镜像的基本镜像，以及您希望在其上安装和运行什么。）如下：  
  
b.编辑这个文件并添加以下内容：

FROM ubuntu:18.04

# Install dependencies

RUN apt-get update && \

apt-get -y install apache2

# Install apache and write hello world message

RUN echo 'hello Docker!' > /var/www/html/index.html

# Configure apache

RUN echo '. /etc/apache2/envvars' > /root/run\_apache.sh && \

echo 'mkdir -p /var/run/apache2' >> /root/run\_apache.sh && \

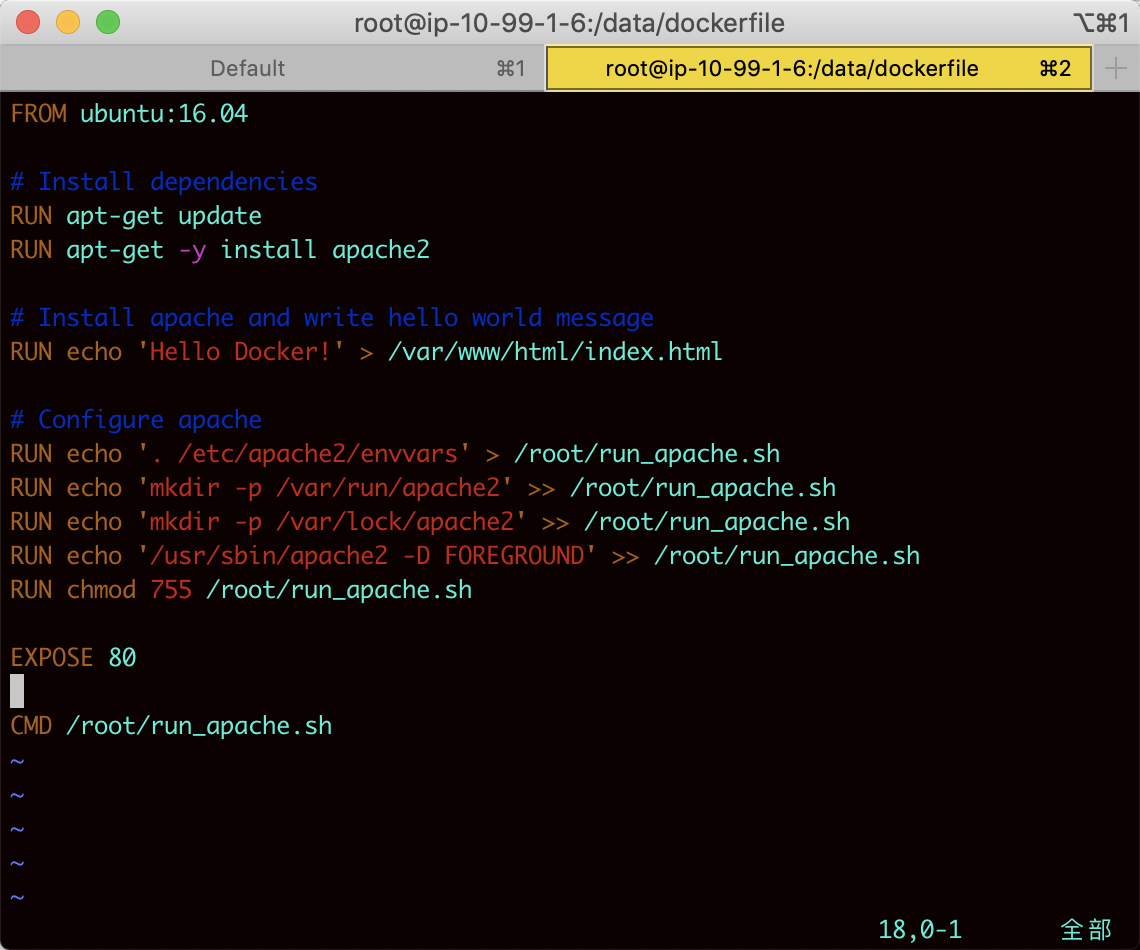
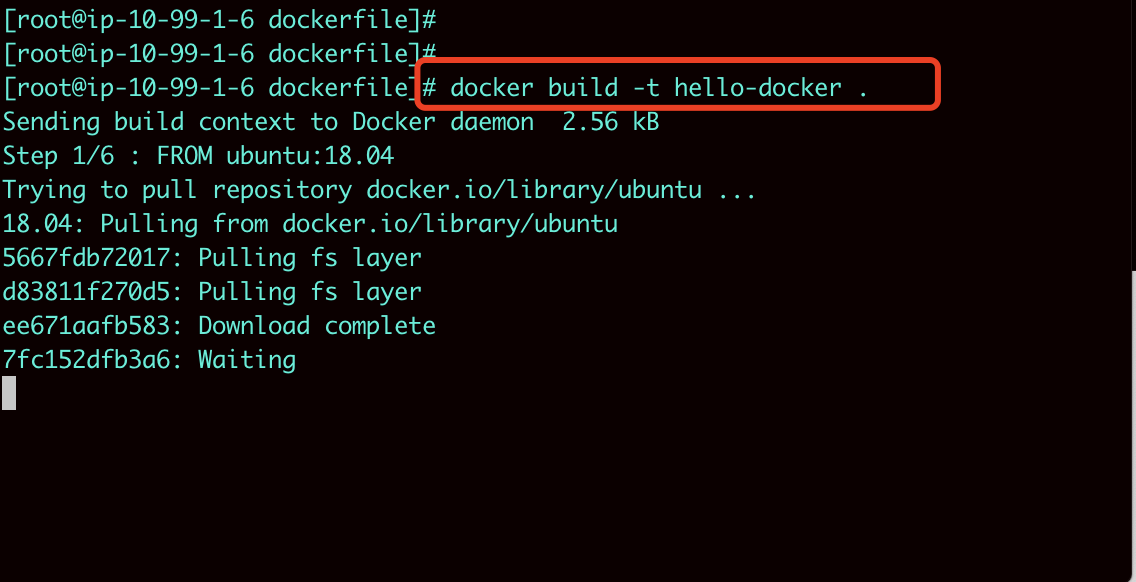
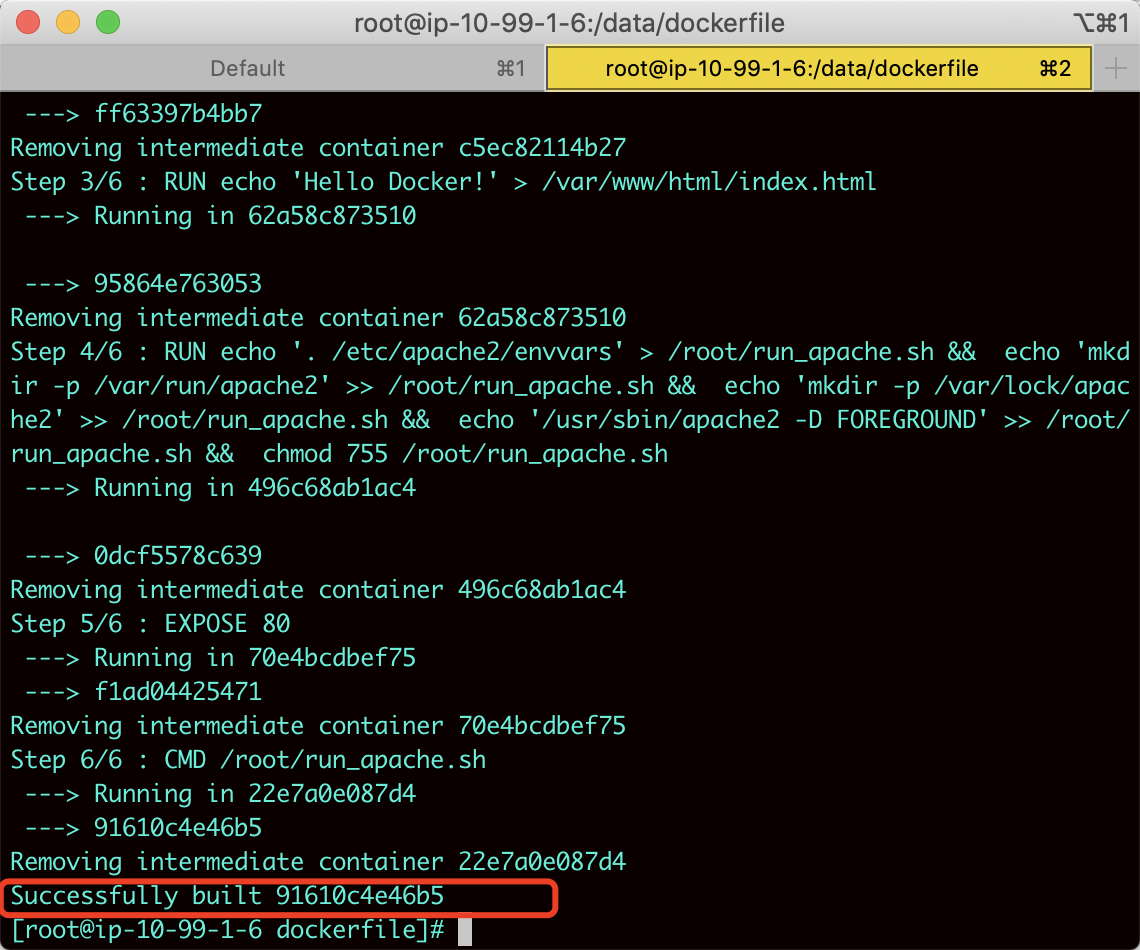
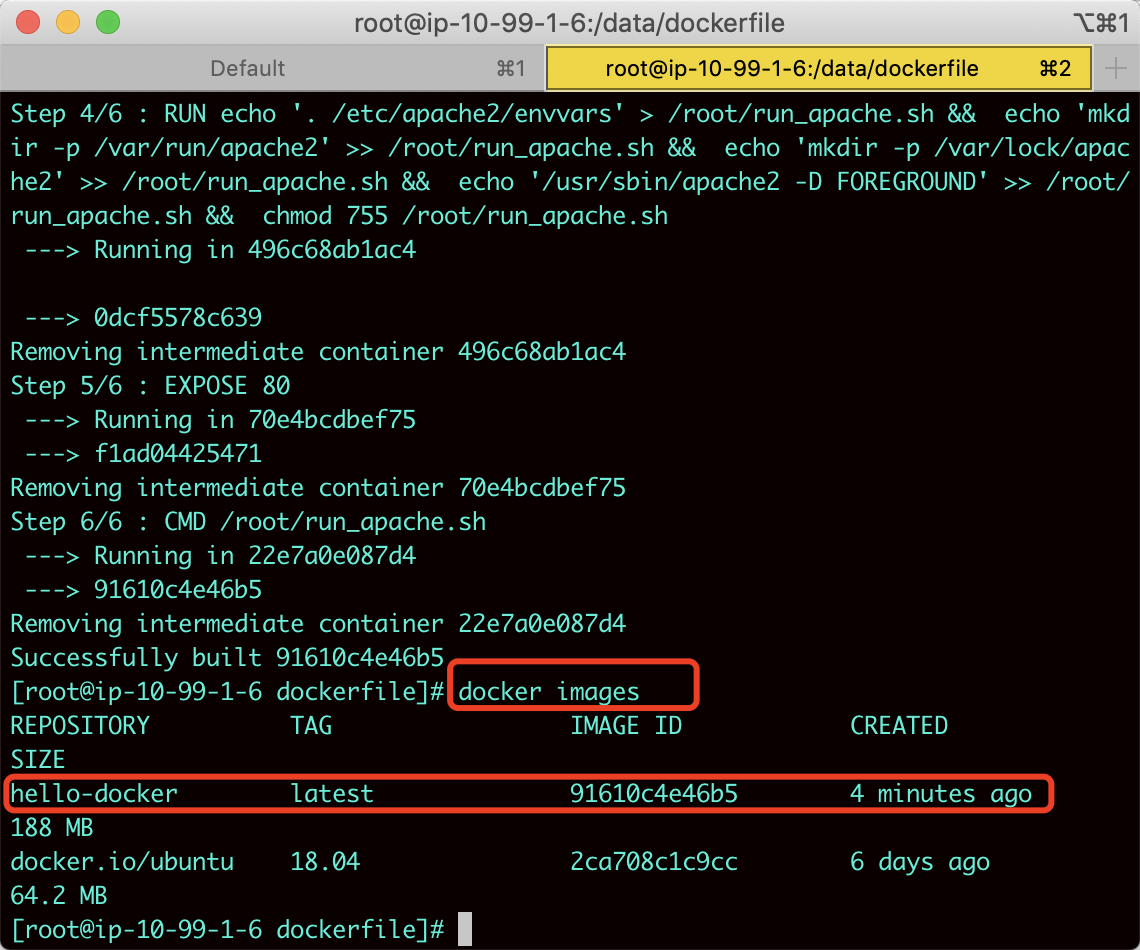
echo 'mkdir -p /var/lock/apache2' >> /root/run\_apache.sh && \

echo '/usr/sbin/apache2 -D FOREGROUND' >> /root/run\_apache.sh && \

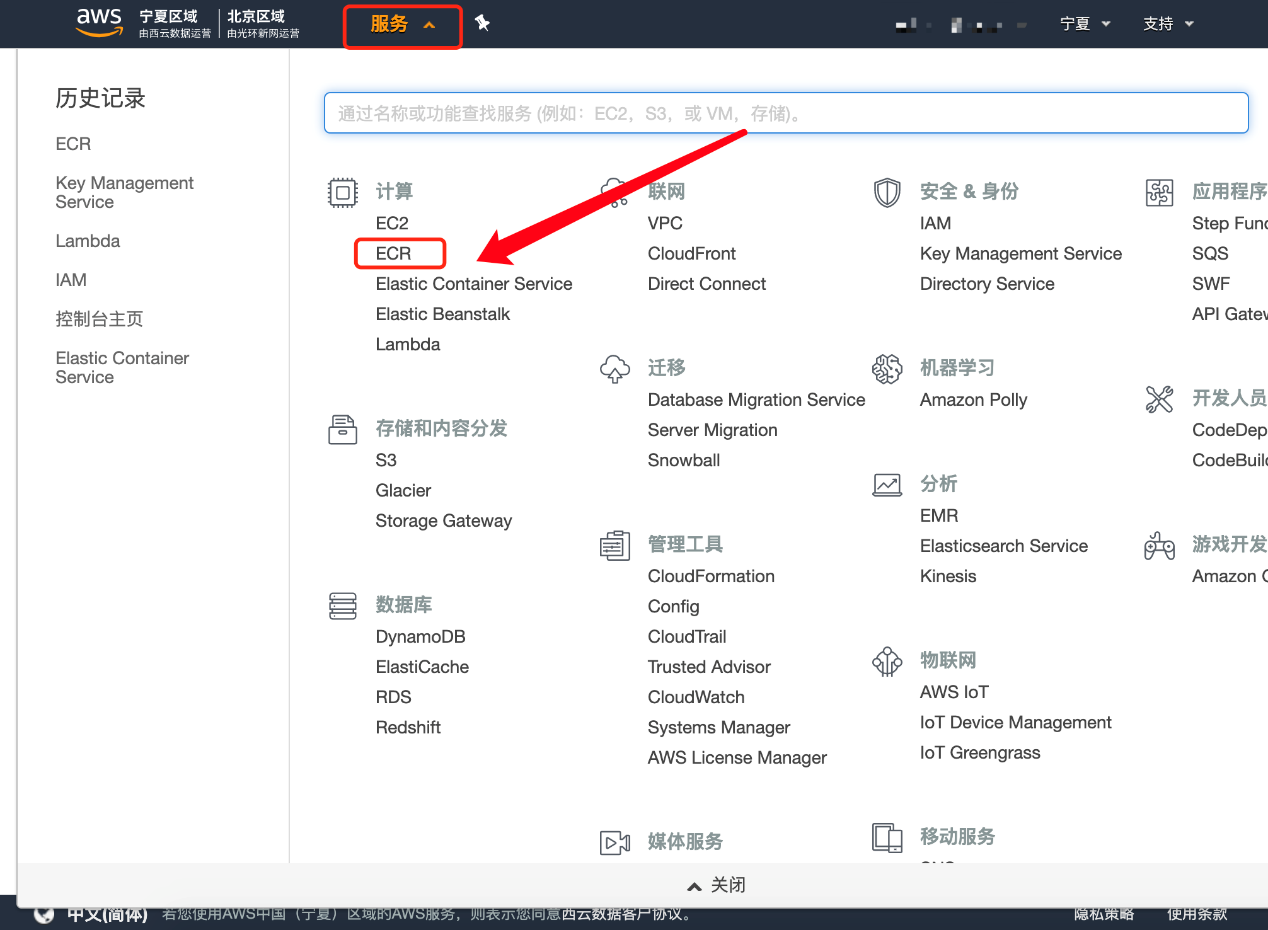
chmod 755 /root/run\_apache.sh

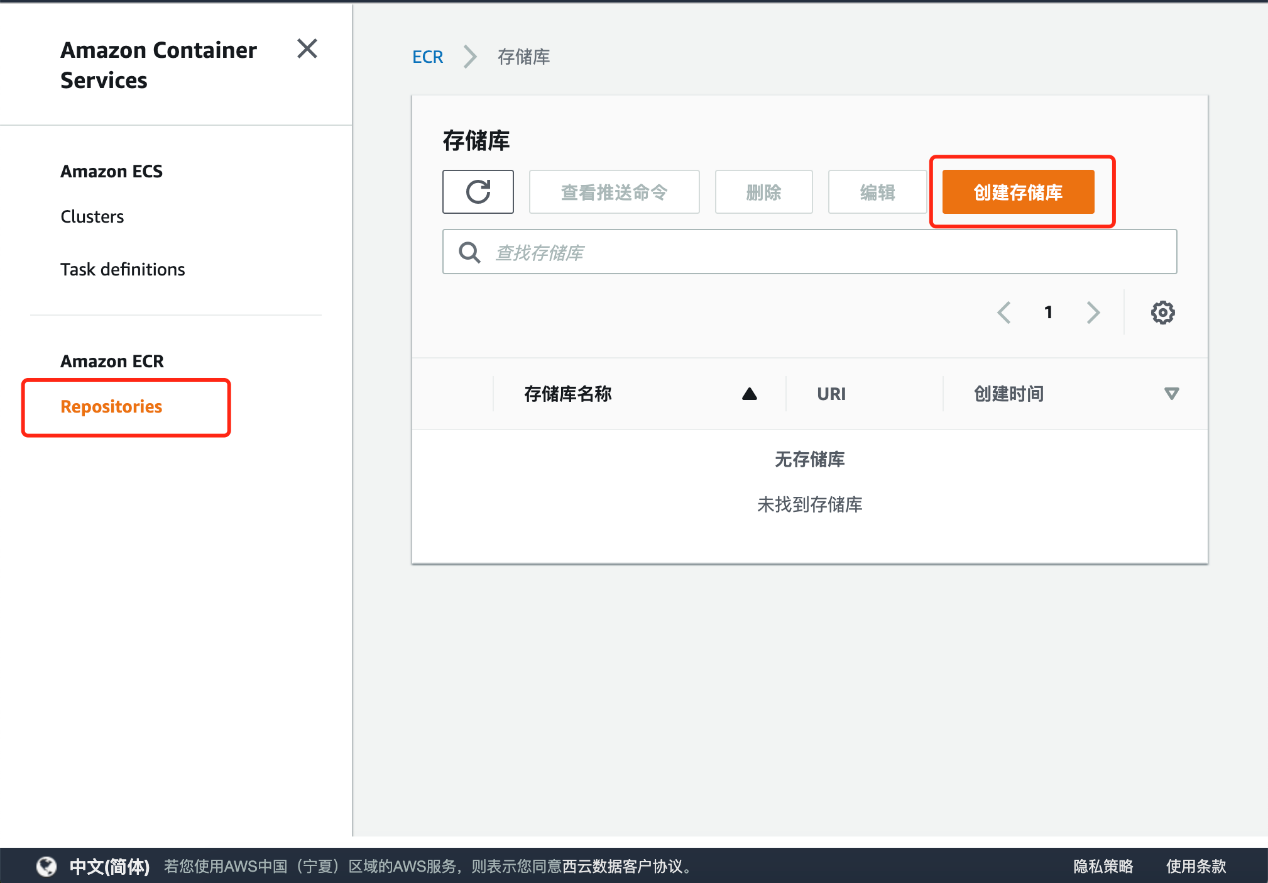
EXPOSE 80

CMD /root/run\_apache.sh

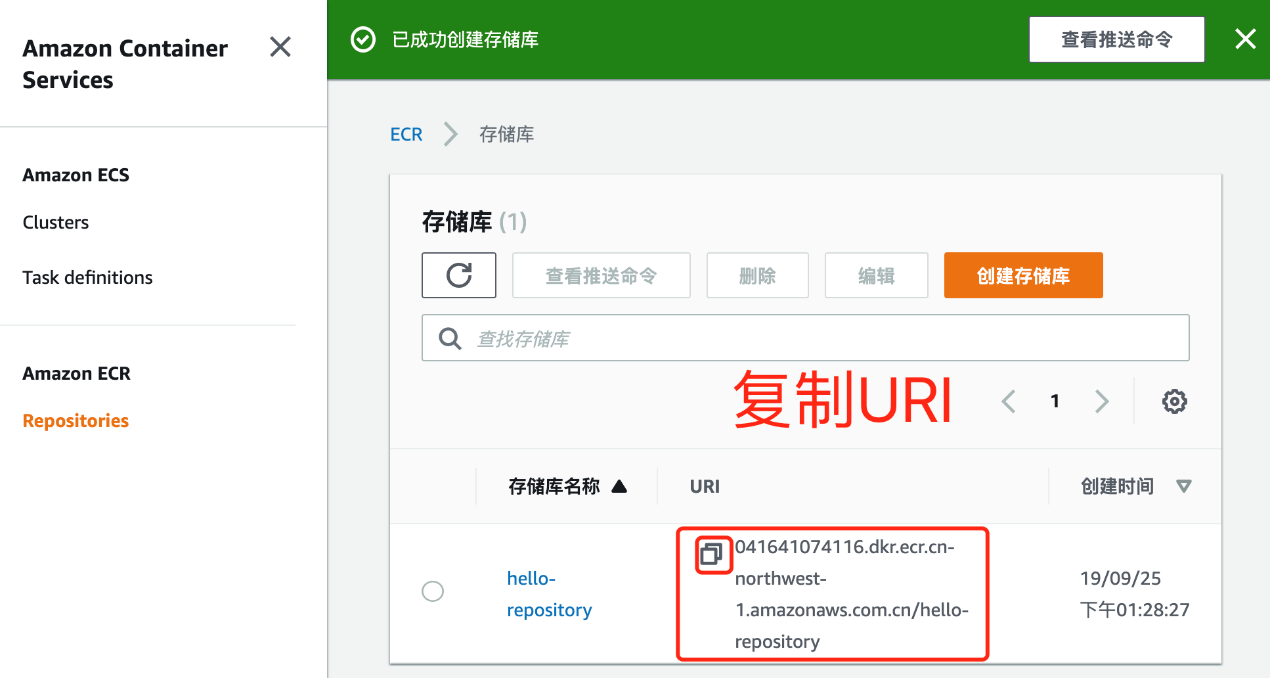
**  
c.从dockerfile构建Docker镜像，如下：  
  
等待镜像构建完成，如下：  
  
d.运行【docker images】命令，验证镜像是否正确创建，如下：  
**

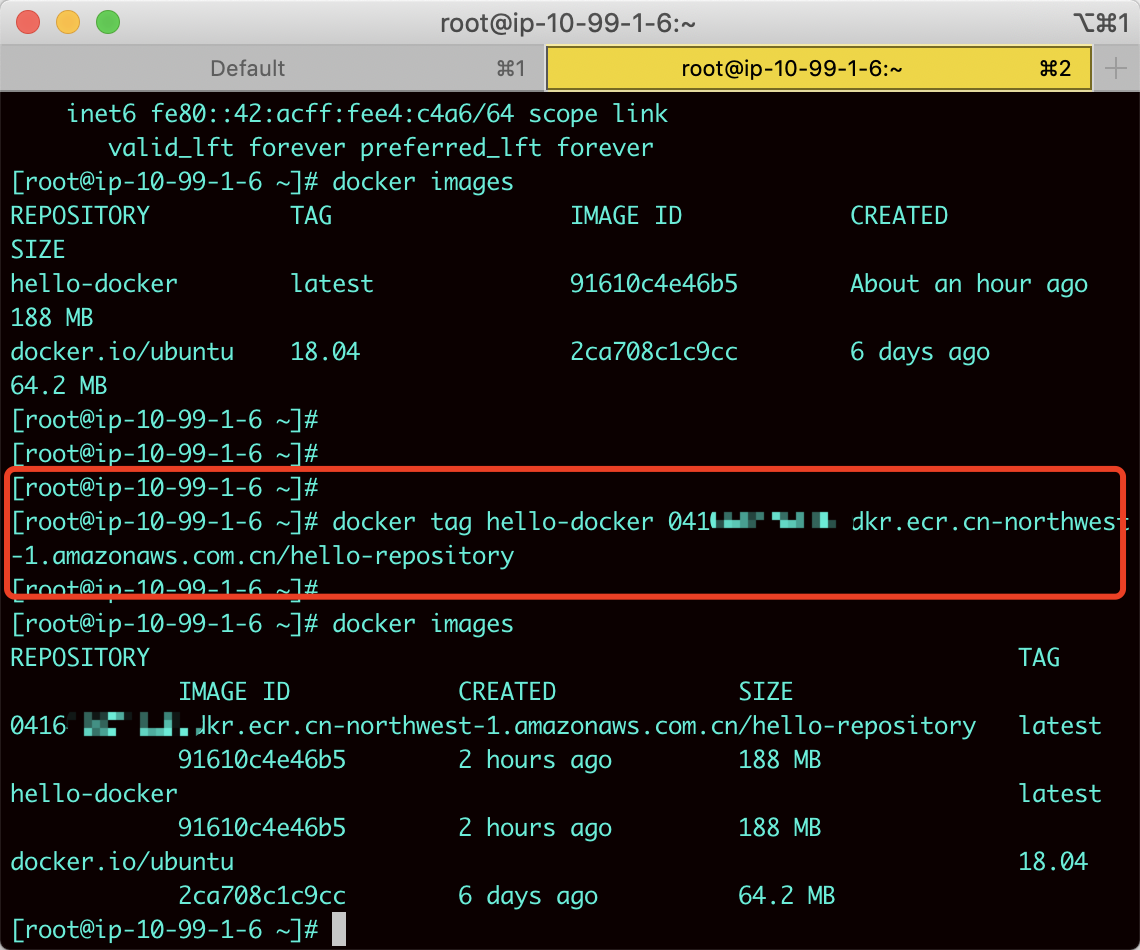
下面我们将把新建的镜像推送到Amazon Elastic Container Registry。

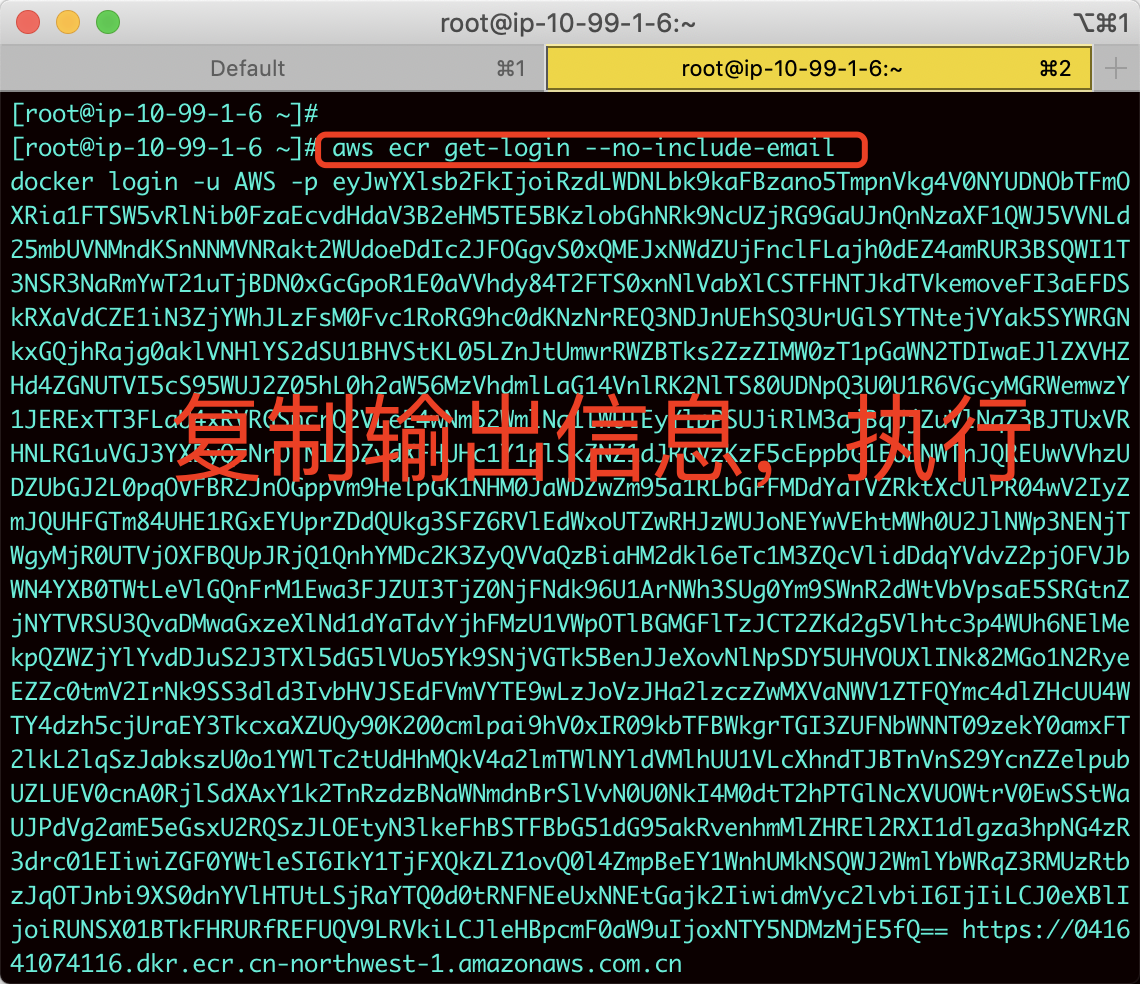
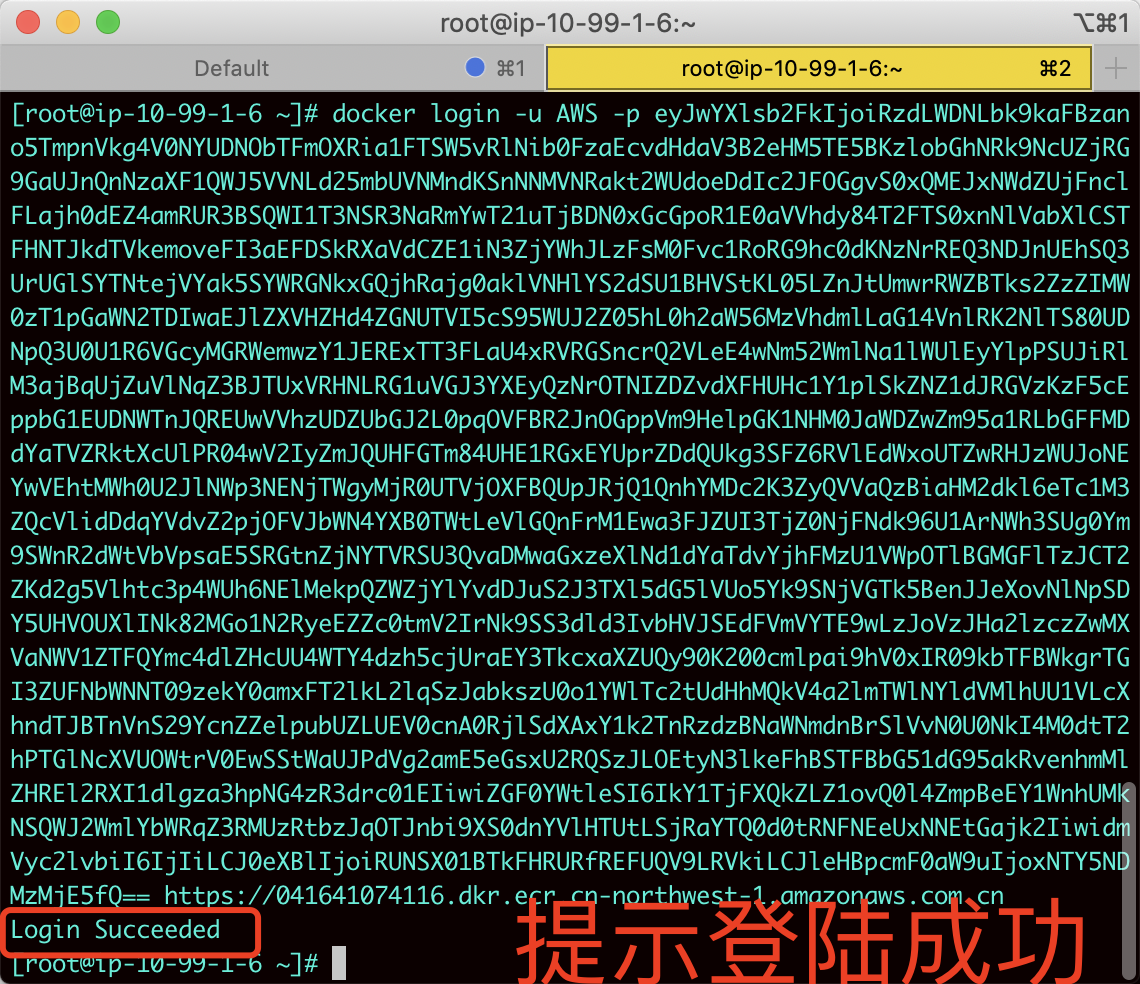
3、点击【服务】，选择【ECR】，如下：  


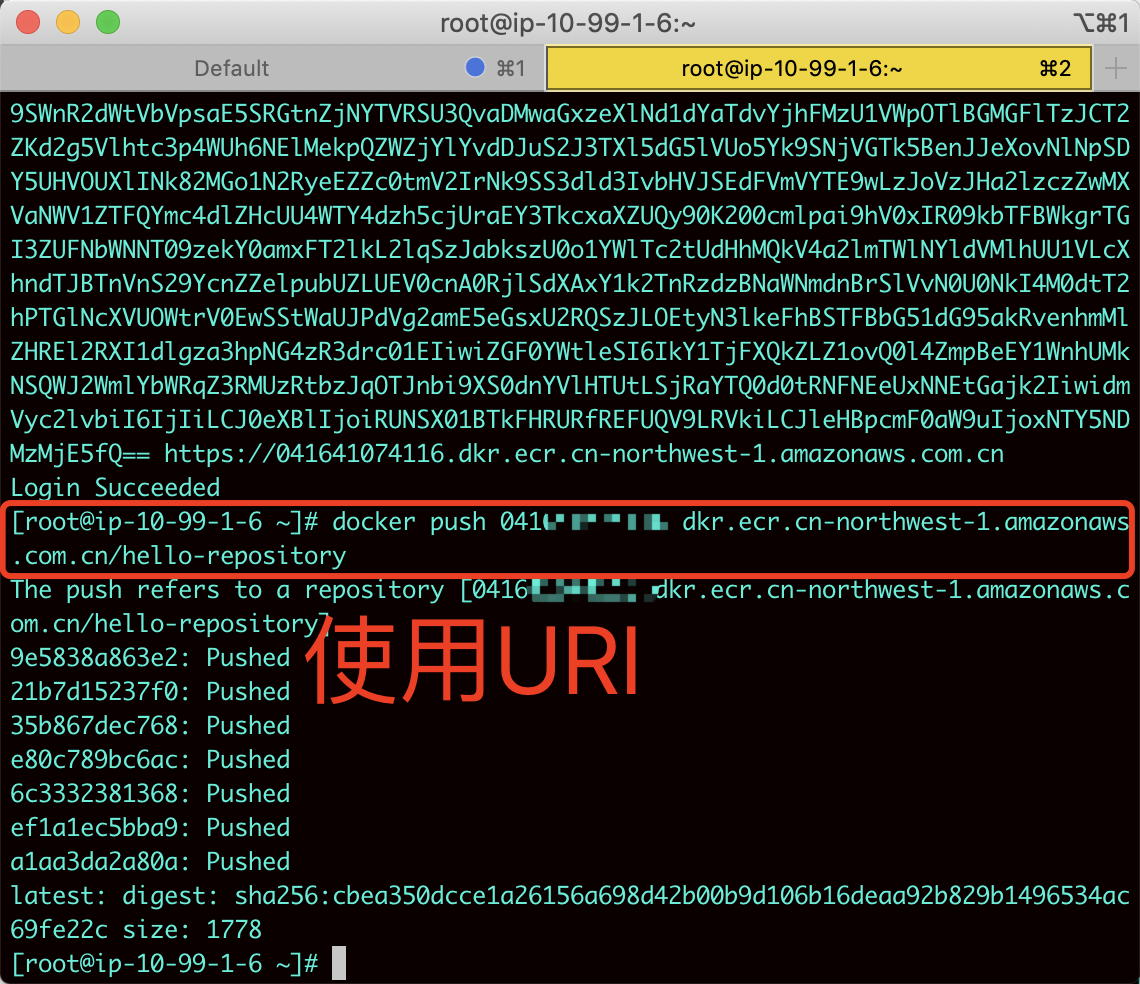
4、创建一个Amazon ECR存储库来存储我们刚刚到【hello-docker】镜像，可以在AWS Console中新建，也可使用命令行创建。本实验使用AWS Console创建一个存储库，在ECR左侧导航窗中选择【Repositories】后再点击【创建存储库】，如下：  


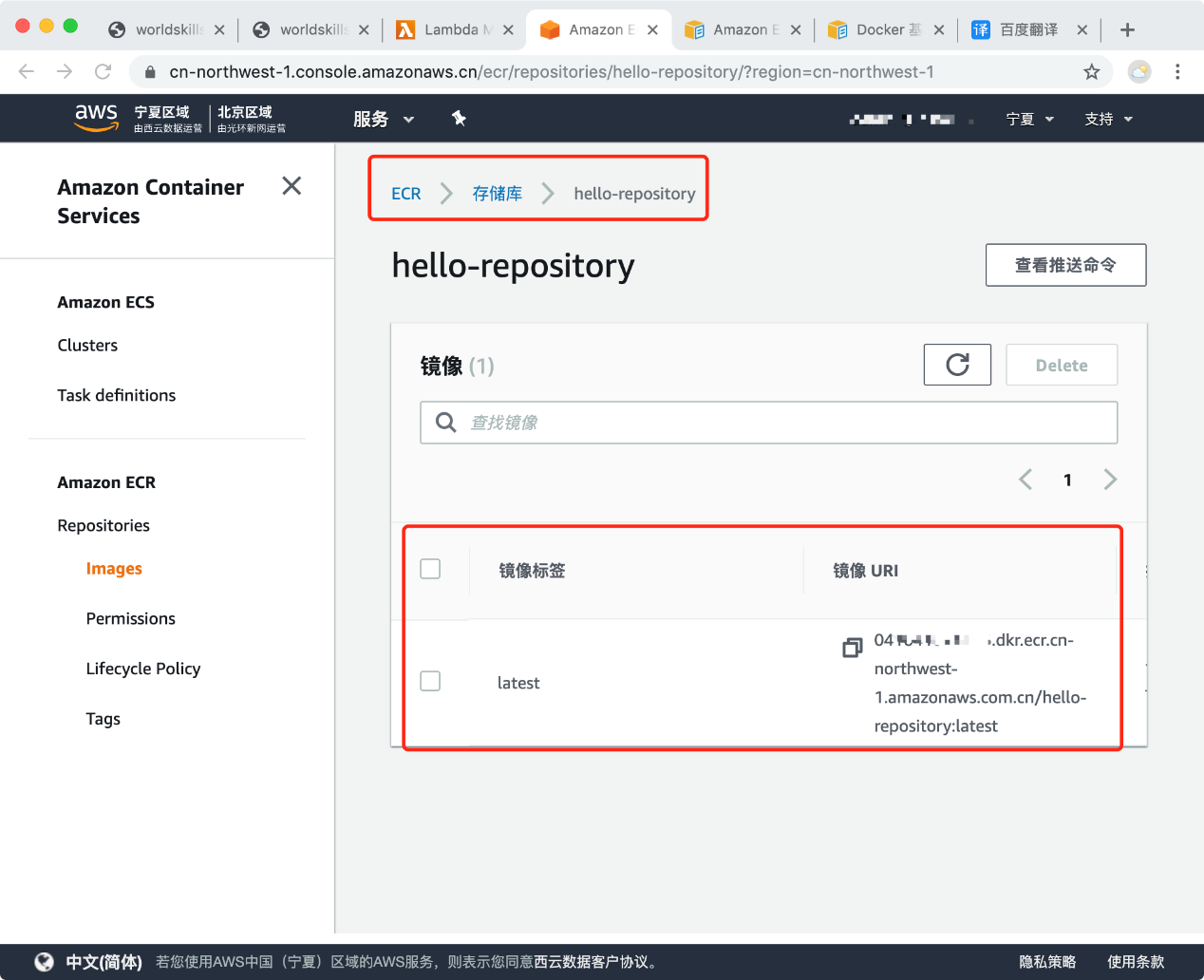
5、在创建存储库页，输入存储库名称后，点击【创建存储库】，如下：  

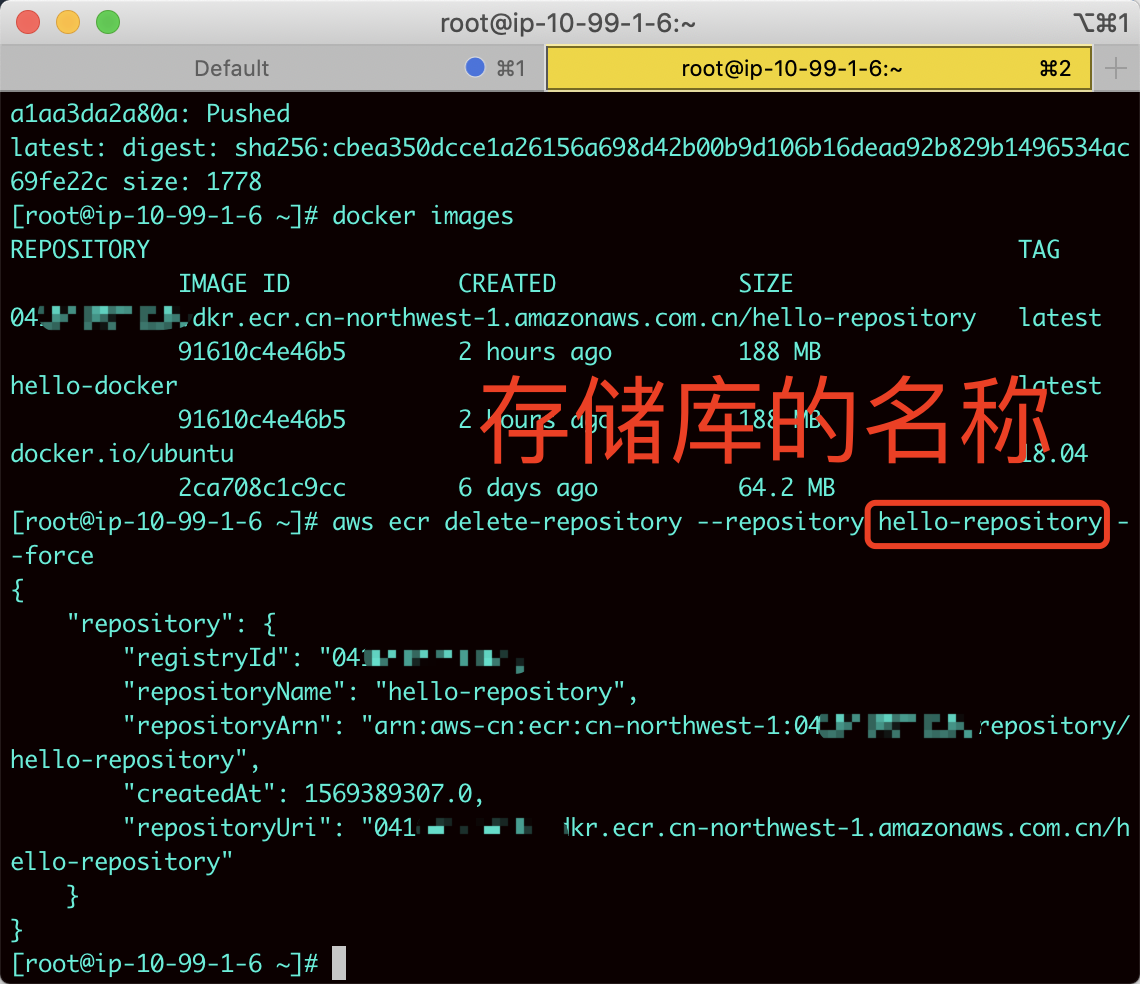

6、创建完成后，复制此URL，等下我们将使用它，如下：  


7、返回到Linux主机，使用刚刚到URI标记刚刚新建的镜像【hello-docker】，如下：  


8、使用【aws ecr get-login –no-include-email 】命令获取docker login 的认证授权命令字符串，如下：  
  


9、使用docker push 命令将镜像上传到Amazon ECR，如下：  


10、在AWS Console中，可查看到我们刚刚push的镜像，如下：  


11、「可选」清理ECR镜像仓库，两种方式，如下：  
        A、第一种，在AWS Console中删除，如下：  
  
  
        B、第二种，使用命令行删除，  


**实验结果**

1、新建存储库，并上传镜像  
