# docker介绍

Docker 是一个开源项目，诞生于 2013 年初，最初是 dotCloud 公司内部的一个业余项目。它基于 Google 公司推出的 Go 语言实现。 项目后来加入了 Linux 基金会，遵从了 Apache 2.0 协议，项目代码在 GitHub 上进行维护。  
  
Docker 自开源后受到广泛的关注和讨论，以至于 dotCloud 公司后来都改名为 Docker Inc。Redhat 已经在其 RHEL6.5 中集中支持 Docker；Google 也在其 PaaS 产品中广泛应用。  
  
Docker 项目的目标是实现轻量级的操作系统虚拟化解决方案。 Docker 的基础是 Linux 容器（LXC）等技术。  
  
在 LXC 的基础上 Docker 进行了进一步的封装，让用户不需要去关心容器的管理，使得操作更为简便。用户操作 Docker 的容器就  
  
像操作一个快速轻量级的虚拟机一样简单。

## docker镜像

docker 镜像（image）就是一个只读模板

docker镜像用来创建docker容器

docker镜像和docker容器的关系，类似于java中class和类与对象之间的关系

## 容器；

docker利用容器（contain）来运行应用

容器就是荣镜像创建的运行实例。他可以被启动、开始停止、删除。每个容器都是相互隔离的、保证安全的平台

可以把容器看成一个建议的linux环境（包括root用户的权限、进程空间、用户空间、网络等）和运行在其中的应用程序

## registry(仓库注册服务器)

registry是集中存放镜像文件的场所

仓库（registry）是对于其中的镜像进行分类管理

**repository：仓库**

一个registry 中会有多个repository

一个repository中会有多个不同的tag 的image

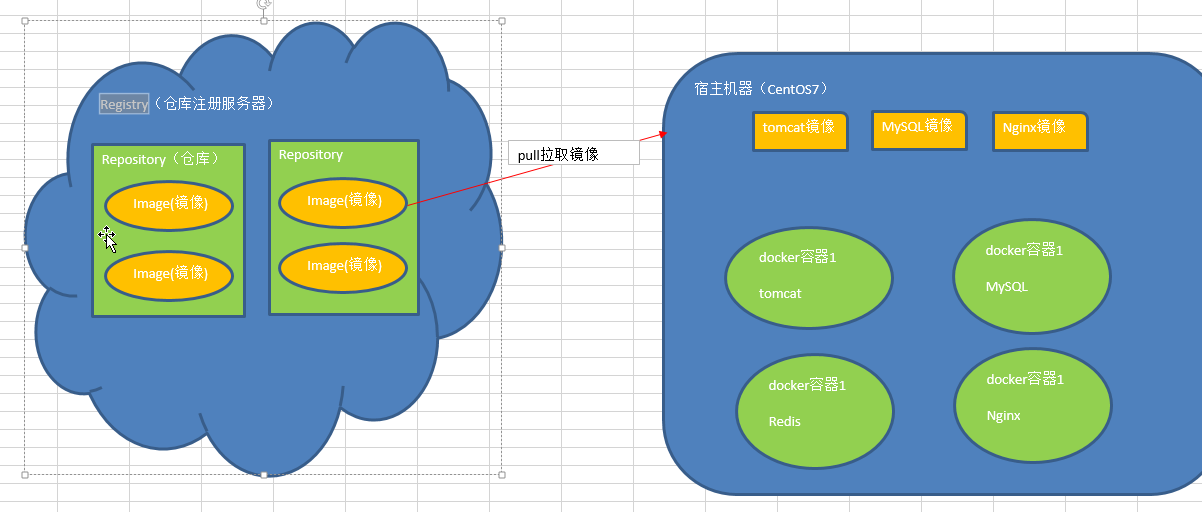
比如在centeros的repository仓库下，有tag 6或7的image镜像

registry分成共有的public 和私有的private两种形式。

最大的共有的registry 是DockHub ，存放了数量庞大的镜像供用户下载使用

国内的公开的registry包括USTC、网易云、DaoCloud、AliCloud等。可以供大陆用户更稳定快捷的访问

用户可以在本地创建一个私有的registry



# docker常用的命令：

## 卸载docker

查看是否安装docker

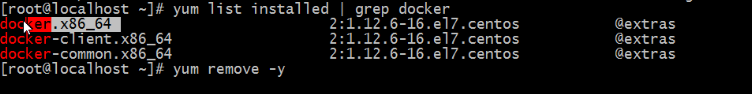
yum list installed | grep docker

如果有安装，需要进行删除

yum remove -y docker.X86\_64

有几个就需要删除一个。

-y表示的不询问这直接删除；可以不写，删除的时候需要进行确认



（资料来源：<http://www.server110.com/docker/201511/11490.html>）

## Docker通常用于如下场景：

web应用的自动化打包和发布；

自动化测试和持续集成、发布；

在服务型环境中部署和调整数据库或其他的后台应用；

从头编译或者扩展现有的OpenShift或Cloud Foundry平台来搭建自己的PaaS环境。

    本文介绍如何在RedHat/CentOS环境下安装Docker。官方文档要求Linux kernel至少3.8以上，且docker只能运行在64位的系统中。由于RHEL6和CentOS6的内核版本为2.6，因此必须要先升级内核。

    下面以CentOS6.5（64位）为例，介绍下docker安装步骤和使用方法：

## 升级内核（带aufs模块）

资料来源：（<http://www.centoscn.com/image-text/config/2015/0718/5877.html>）

1.下载安装

在centos上运行docker必须是64位并且内核在2.6.32-431或者更高内核版本，建议完全更新系统

[root@iZ25gswv4olZ ~]# getconf LONG\_BIT ---查看位数32/64

64

[root@iZ25gswv4olZ ~]# uname -r --查看内核版本

2.6.32-431.23.3.el6.x86\_64

[root@iZ25gswv4olZ ~]# yum update ---完全更新系统

[root@iZ25gswv4olZ ~]# uname -r

2.6.32-431.23.3.el6.x86\_64

[root@iZ25gswv4olZ ~]# lsb\_release -a ----列出所有版本信息

LSB Version: :base-4.0-amd64:base-4.0-noa

## 安装docker

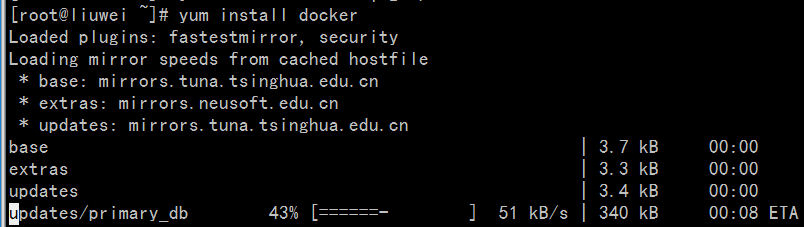
### 安装过程

需要在联网的情况下安装

yum -y install docker

或者

yum -y install docker-io

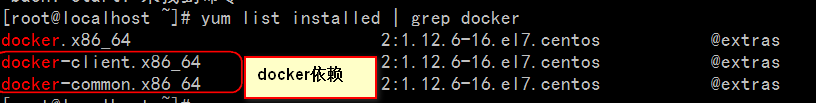


直接进行下载安装了

### 查看安装的docker

yum list installed | grep docker

查看已经装的docker

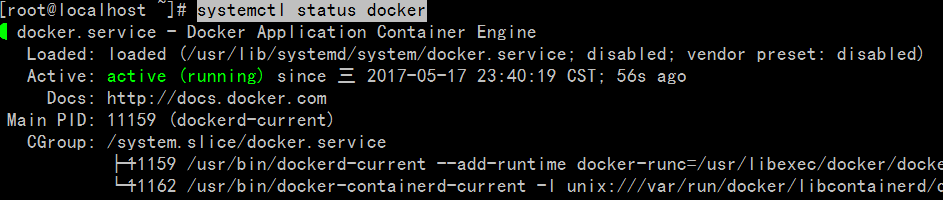


### 启动docker

systemctl start docker

### 查看docker状态

systemctl status docker



启动成功，可以使用docker

### 启动,停止，重启docker

**停止docker**

systemctl stop docker

**启动docker**

systemctl start docker

或者

systemctl start docker.service

**重启docker**

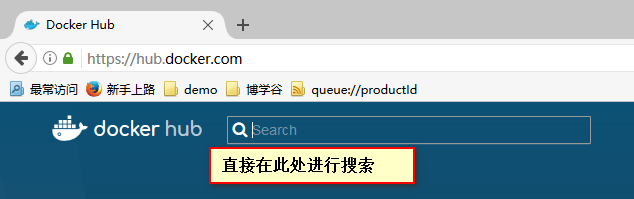
systemctl stop docker

注意：systemctl 也可以换成service

这两个是相同的效果

## 官网

拉取docker 的网址：<https://hub.docker.com/>



比如搜索mysql的 的结果，分成不同的版本；



### 从docker hub 拉取

将Docker Hub 仓库中下载一个 CentOS 7版本的操作镜像

docker pull centos:7

这是docker 默认的公用的仓库，不过缺点是国内下载会比较的慢

## 从sutc 拉取（建议使用）

在宿主机器编辑文件：

vi /etc/docker/daemon.json

请在该配置文件中加入（没有该文件，请创建一个文件）；

{

"registry-mirrors": ["https://docker.mirrors.ustc.edu.cn"]

}

最后，需要重启docker服务

systemctl start docker.service

之后再使用pull 命令拉去镜像，这时候是从ustc 获取镜像。而且速度很快；

阿里云加速说明

<https://cr.console.aliyun.com/#/accelerator>

## 直接拉取

下载mysql 中5.5 镜像的指令是

docker pull mysql:5.5

## 获取和删除

导入mysql，

docker pull mysql:5.5



### 列出镜像

docker images

### 删除镜像

docker rmi mysql:5.5

或者是（使用id进项删除）

docker rmi 33efe4d711c7

### 导入导出镜像

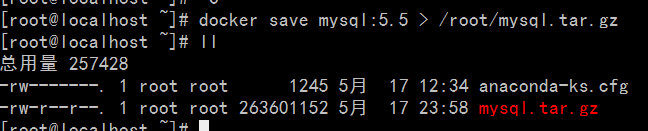
导入镜像:docker save 镜像 > /root/xx.tar.gz

镜像表示的 仓库 + 版本号

或者是 id

例如：需要时间。导出

docker save mysql:5.5 > /root/mysql.tar.gz



这样就导出了mysql 的镜像了

相反的过程

docker load < /root/mysql.tar.gz

# docker 容器

启动容器：

以交互的方式启动容器：docker run -it --name 容器的名称，镜像

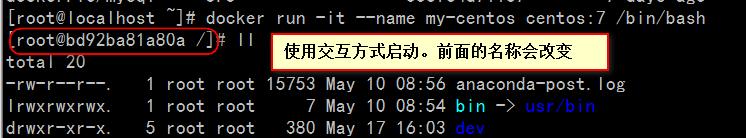
/bin/bash

以守护进程的方式启动容器：docker run -d –name 容器的名称镜像

## 交互方式启动

使用容器启动centeros 7

docker run -it --name my-centos centos:7 /bin/bash



退出命令：exit

## 守护线程启动

启动命令

注意：启动的名称不可以一致

docker run -d --name my-centos2 centos:7



表示启动成功

此时的依然是使用宿主启动命令

## 保存进行

docker commit 23c18d958279 bigdata:v0.2

资料来源： <https://www.jianshu.com/p/06f1705f8840>

## 检查会否正在运行 docker

查询正在运行的程序；

docker ps

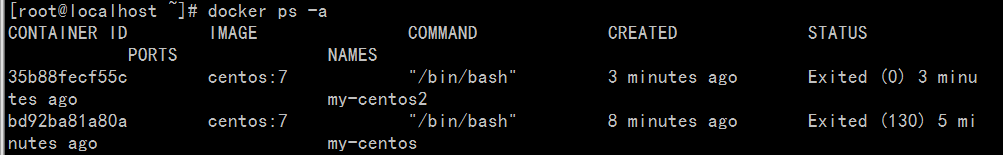
可以使用，查询所有的程序，包括已经停止的

docker ps -a

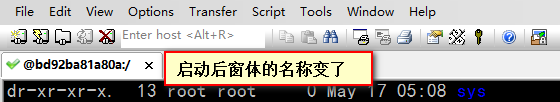
表示查询所有

docker ps -l

表示最近运行的程序



上面



## 停止

语法

docker stop 启动的名称

比如上面的

## 删除

删除一个已经停止的；（后面是名称或者是id）

docker rm my-centos

# 启动tomcat案例

## 启动tomcat

（资料来源：<http://blog.csdn.net/q26335804/article/details/47806105>

公司要把j2ee的project搬到 [**Docker**](http://lib.csdn.net/base/docker)里，所以，先从一个tomcat的镜像开始吧！

下载tomcat

docker pull tomcat:7

查看现有的镜像

docker images

基于这个image，创建一个容器吧

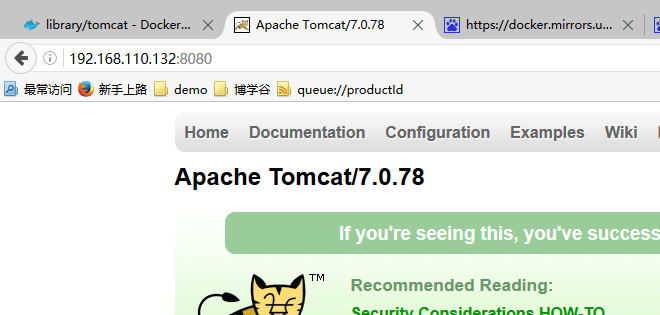
docker run -d --name dev\_tomcat -p 8080:8080 tomcat:7

// --name 给这个容器起一个名字

// -p host到container的端口映射

-p 后面第一个端口号表示访问的地址：

第二端口好表示的 ；tomcat的端口号



启动后的效果。使用浏览器进行访问

## 部署tomcat

### 进入交互

docker exec -it dev\_tomcat /bin/bash

红色表示容器的名称。根据实际情况进行改变

* **docker run：启动container**
* **ubuntu：你想要启动的image**
* **-t：进入终端**
* **-i：获得一个交互式的连接，通过获取container的输入**
* **/bin/bash：在container中启动一个bash shell**

（资料来源： <https://blog.csdn.net/dongdong9223/article/details/52998375> ）



进入交互以后，界面就改变了，不是直接的命令行

查看当前目录pwd

目录下的文件ls　没有ll

退出exit

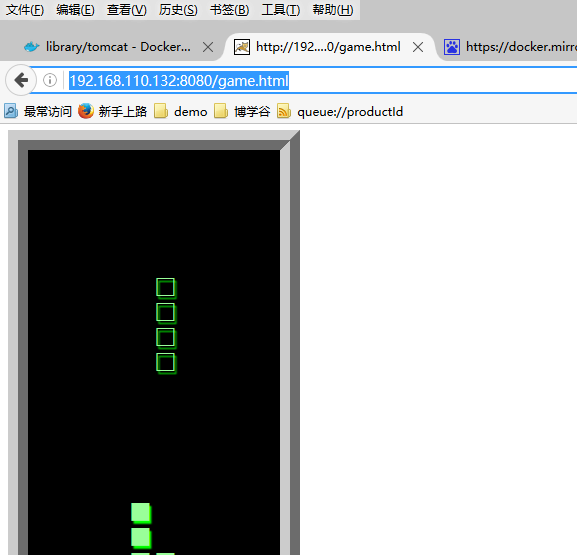
### 部署



docker cp game.html dev\_tomcat:/usr/local/tomcat/webapps/ROOT

把指定的文件拷贝到指定的位置

docker cp 文件的名称 docker容器 冒号 路径



比如上面是把一个html 页面 部署到tomcat 总root 目录下。

访问的地址：

http://192.168.110.132:8080/game.html