**docker概述**

概述

什么是容器

容器技术已经成为用用程序封装和交付的核心技术

容器技术的核心有以下几个内核技术组成：

CGroups（control groups）-资源管理，限制容器使用的资源

namespace-进程管理

selinux安全

由于是在物理机上实时隔离，启动一个容器，可以像启动一个进程一样快速

什么是docker

docker是完整的一套容器管理系统

docker提供了一组命令，让用户更加方便直接地使用容器技术，而不需要过多关心低层内核技术

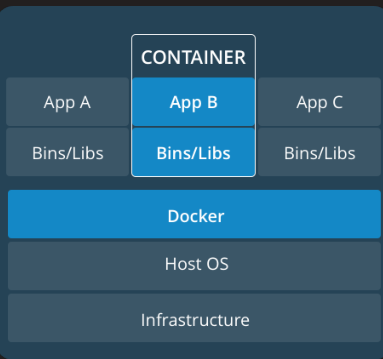
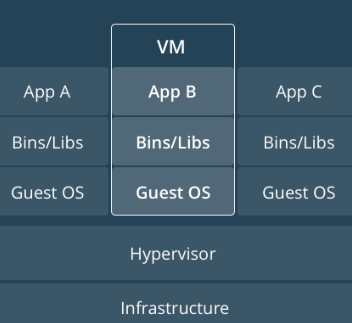
docker特性

docker优点

相比于传统的虚拟化技术，容器更加简洁高效

传统虚拟机需要给每个VM安装操作系统

容器使用的共享公共库和程序



docker的缺点

容器的隔离性没有虚拟化强

共用linux内核，安全性有先天缺陷

selinux难以驾驭

监控容器和容器排错是挑战

**部署docker**

安装软件

安装前准备

需要64位操作系统

至少RHEL6.5以上的版本，强烈推荐RHEL7

关闭防火墙（不是必须）

安装docker

软件包列表：

docker-engine

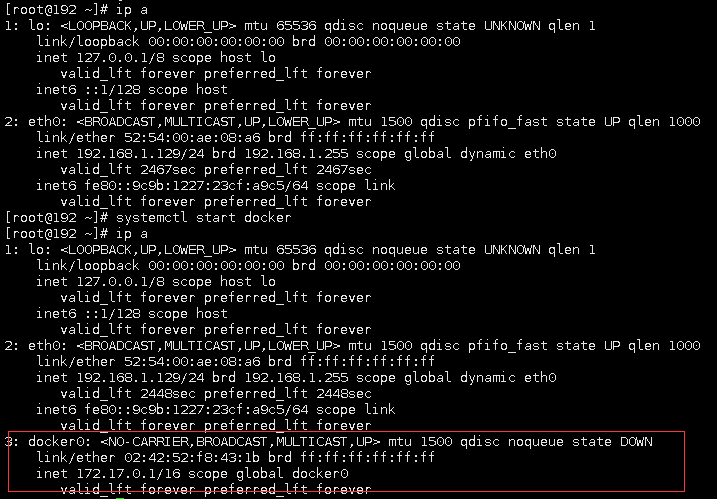
docker-engin-selinux

rpm -ivh docker-engine-1.12.1-1.el7.centos.x86\_64.rpm

rpm -ivh docker-engine-selinux-1.12.1-1.el7.centos.noarch.rpm

systemctl start docker

systemctl enable docker



**docker镜像**

基本概念

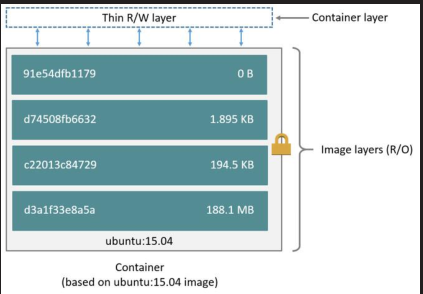
什么是镜像

在docker中容器是基于镜像启动的

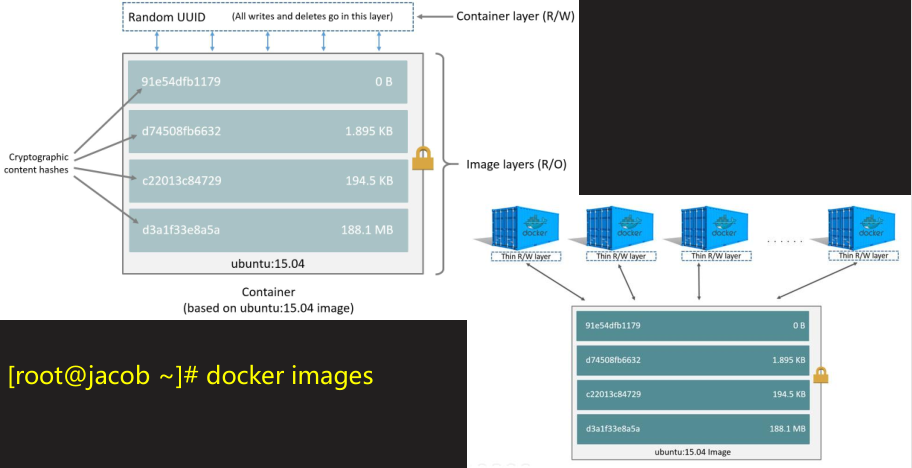
镜像是启动容器的核心

镜像采用分层设计

使用快照的COW技术，确保底层数据不丢失



前端容器，后端镜像，一个后端盘可以创建多个容器，就像一个后端盘可以创建多个虚拟机，后端盘用镜像封装而成，才用分层设计



docker hub镜像

<https://hub.docker.com>

docker官方提供公共镜像的仓库（registry）

docker search rhel7

docker search centos

docker search nginx

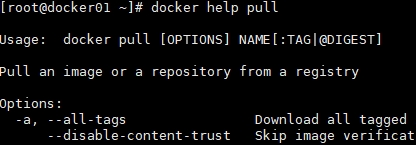
docker search mysql

镜像操作

下载、上传镜像

下载镜像（从镜像仓库中下载镜像）

docker help 命令

查看pull的用法

docker pull rhel7

上传镜像（上传镜像到仓库）

docdocker help push

docker push rhel7

导入导出操作

导入镜像（通过tar包文件导入镜像）

docker load < xx.tar

docker images

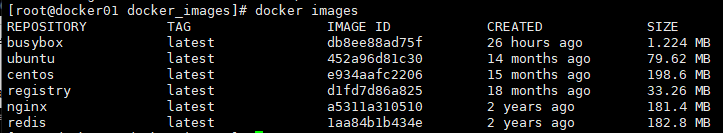












导出镜像（将本地镜像导出为tar包）

docker images

docker save image\_name:tag > xx.tar





启动镜像

启动centos镜像生成一个容器

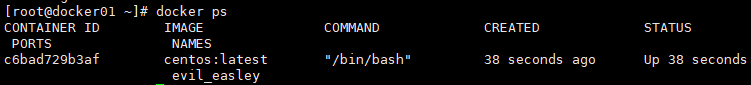
docker images

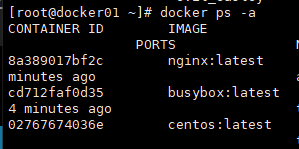
docker run -it 镜像名字:标签 镜像里的命令（i代表交互式的，t代表终端，不写代表默认，服务需要制定）



开启另一个终端（查看容器信息）

docker ps



查看所有

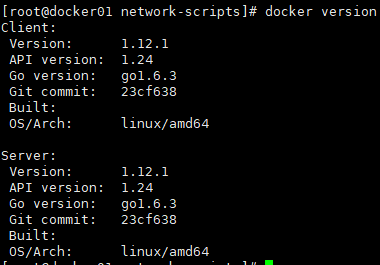
**docker基本命令**

镜像常用命令

命令列表

命令列表

docker version //查看版本



docker images //查看镜像列表

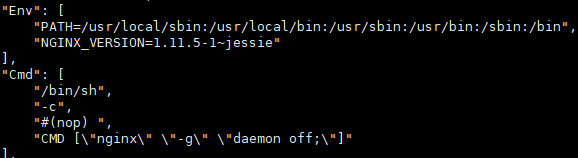


docker history 镜像名//查看镜像制作历史



docker inspect //查看详细信息





docker pull //下载镜像







docker push //上传镜像

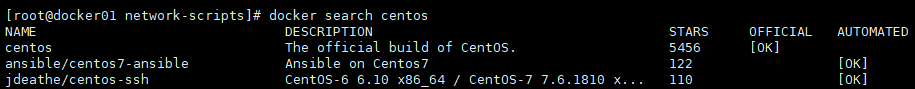
docker rmi //删除本地镜像

如果报错，说明有容器存在，先删除容器

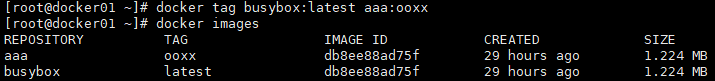
docker save//镜像另存为tar包

docker load //使用tar包导入镜像

docker search 关键字 //搜索镜像



docker tag //创建连接



docker images

查看镜像列表

镜像仓库名称

镜像标签

镜像ID

创建时间

大小

docker history

查看镜像历史

了解镜像制作过程

详细参考后面的dockerfile内容

docker inspect

查看镜像底层信息

了解镜像环境变量、存储卷、标签等信息

docker rmi

删除本地镜像

注意：启动容器时删除镜像会提示错误

docker save|load

保存本地镜像另存为tar文件

方便其他人使用tar包导入镜像

使用tar包导入镜像

docker tag

重命名镜像名称（复制）

容器常用命令

命令列表

docker run //运行容器

docker ps //查看容器列表

docker stop//关闭容器

docker start //启动容器

docker restart //重启容器

docker attach|exac //进入容器

docker inspect //查看容器底层信息

docker top //查看容器进程列表

docker rm //删除容器

docker run

使用镜像启动容器

-i 交互式

-t 终端

-d 后台进程

--name 指定名字

启动一个 交互式的终端/bin/bash

docker run -it centos /bin/bash



启动一个服务（非交互）nginx

docker run -d nginx



启动一个交互式的终端，放在后台运行/bin/bash &

docker run -itd centos /bin/bash



指定名称



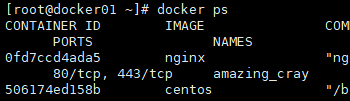




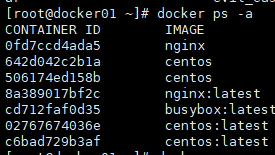
docker ps

列出容器列表

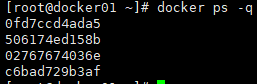
docker ps查看正在运行的容器



docker ps -a 查看所有容器列表



docker ps -q 仅显示容器ID



docker stop|start|restart

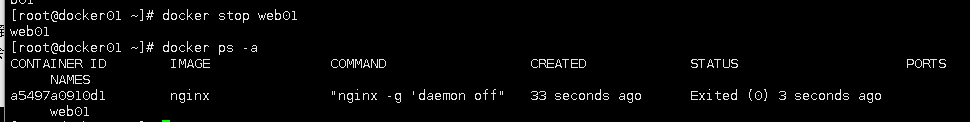
管理容器

docker stop 关闭容器（支持使用docker命令结果）



docker start开启容器







docker restart重启容器

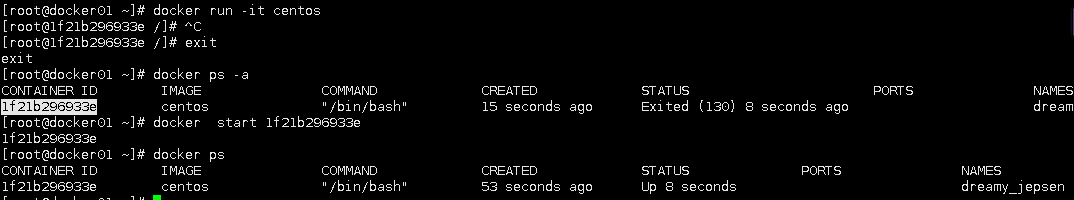


docker attach|exec

进入容器

docker attach 进入容器，exit会导致容器关闭，Ctrl+p+q是放到后台

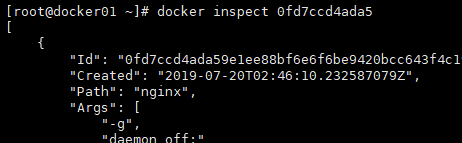
docker exec 进入容器，退出时不会关闭容器，重新启动一个命令





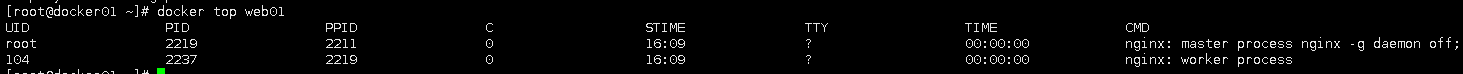
docker inspect

查看容器底层信息



docker top（相当于在容器内部执行ps）

查看容器进程列表



docker rm

删除容器

注意，删除正在运行的容器时会提示错误



