

docker简介与安装

<https://docs.docker.com/engine/docker-overview/#docker-architecture>

<https://www.qikqiak.com/k8s-book/docs/2.Docker%20简介.html>

Securely build, share and run modern applications anywhere

docker是一个用于开发、交付、运行应用程序的平台。开发人员可以通过Docker将应用程序变成一种标准化的、可移植的、自管理的组件。docker封装的镜像可以运行在所有安装docker的服务器上。

docker简介

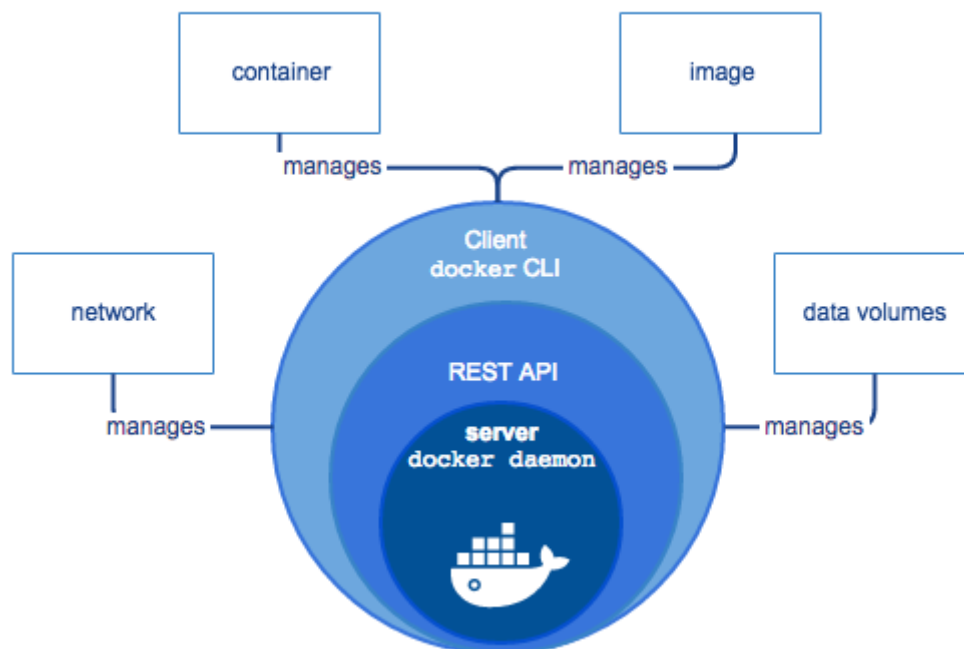
docker与虚拟机的区别

相比而言，docker更加轻量级，更加方便使用

1. docker构建在操作系统层面；而虚拟机依赖物理CPU和内存，是硬件级别的。
2. docker引入类似源码管理的机制，将容器的快照历史版本一一记录，便于切换。
3. docker可以通过dockerfile文件(可手动编写)来构建整个容器，能通过发布dockerfile来定义应用环境和依赖。

Docker Engine

docker engine 是一个C/S架构的应用程序，如下图：

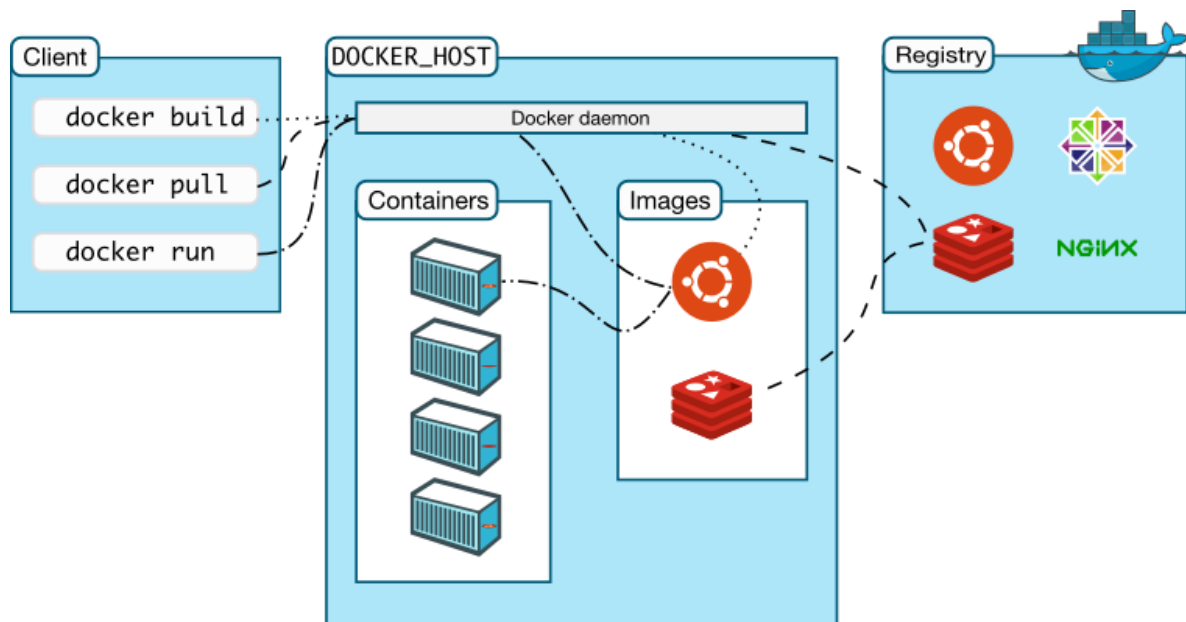


docker架构

Docker使用C/S(客户端/服务器)体系的架构，Docker客户端与Docker守护进程通信，Docker守护进程负责构建，

运行和分发Docker容器。Docker客户端和守护进程可以在同一个系统上运行，也可以将Docker客户端连接到远程

Docker守护进程。Docker客户端和守护进程使用rest api通过Unix套接字或网络接口进行通信。



docker基础元素

镜像：本质上是一个压缩包文件，类似于虚拟机镜像

容器：可以认为是一个正在运行中的镜像，实际上只在镜像文件的基础上添加了一层读写层。

docker仓库

仓库：集中保存镜像文件的地方，提供下载上传等管理镜像文件的操作

docker仓库分公有仓库(<https://hub.docker.com/>)与私有仓库(192.168.100.156)

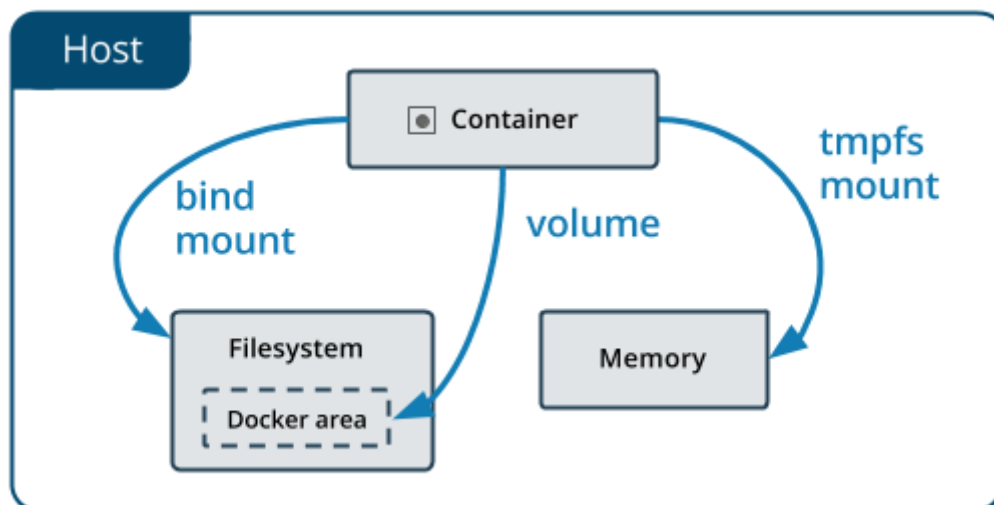
数据共享与持久化

数据持久化(将容器中的文件存储到主机上面)一般有三种方法，volumes、bind mounts、tmpfs mount。

volumes：它会将文件存储在docker管理的文件夹里面(/var/lib/docker/volumes/ on Linux), 同时也是docker里面管理数据最好的一种方式。

bind mounts：相当于挂载，可以将文件保存在主机的任何一个地方

tmpfs mounts：仅存储在主机系统的内存中，并且永远不会写入主机系统的文件系统中。



docker网络

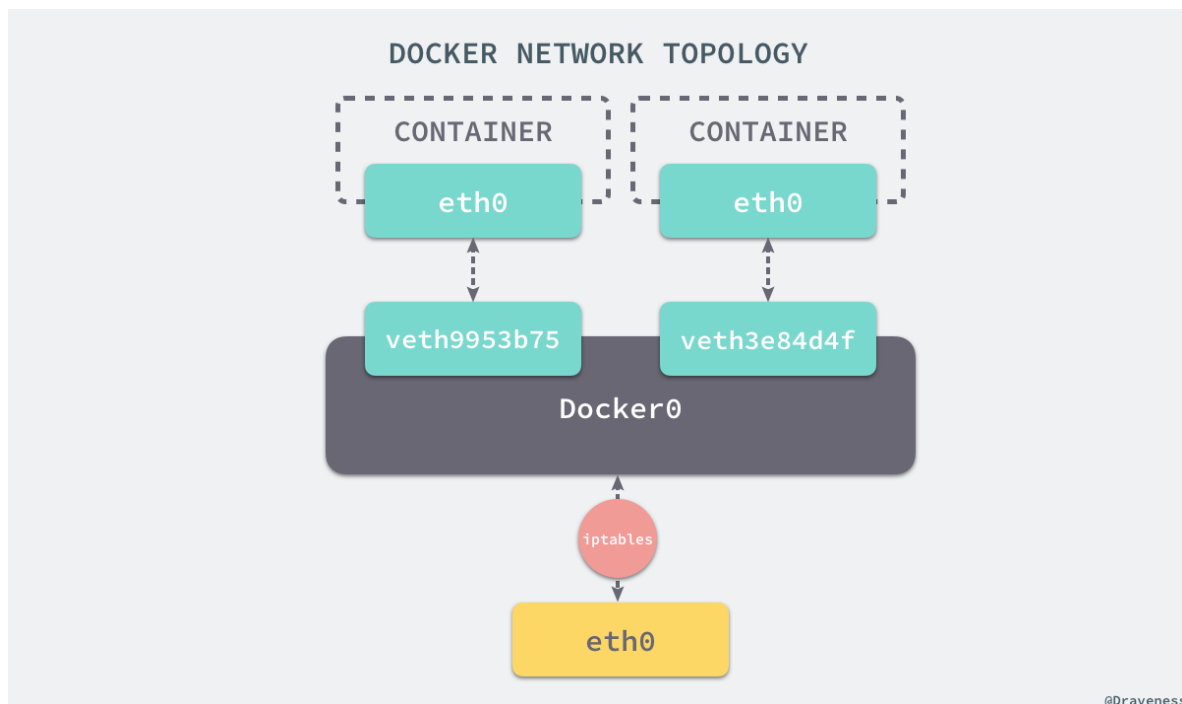
docker为我们提供了四种不同的网络模式，Host、Container、None 和 Bridge 模式。

默认采用Bridge模式：当 Docker 服务器在主机上启动之后会创建新的虚拟网桥 docker0，

随后在该主机上启动的全部服务在默认情况下都与该网桥相连。docker0 会为每一个容器分配一个新的 IP

地址并将 docker0 的 IP 地址设置为默认的网关。网桥 docker0 通过 iptables 中的配置与宿主机上的网卡相连，

所有符合条件的请求都会通过 iptables 转发到 docker0 并由网桥分发给对应的机器。



镜像构建

镜像创建可以通过Dockerfile和docker commit [OPTIONS] CONTAINER [REPOSITORY[:TAG]]来创建。

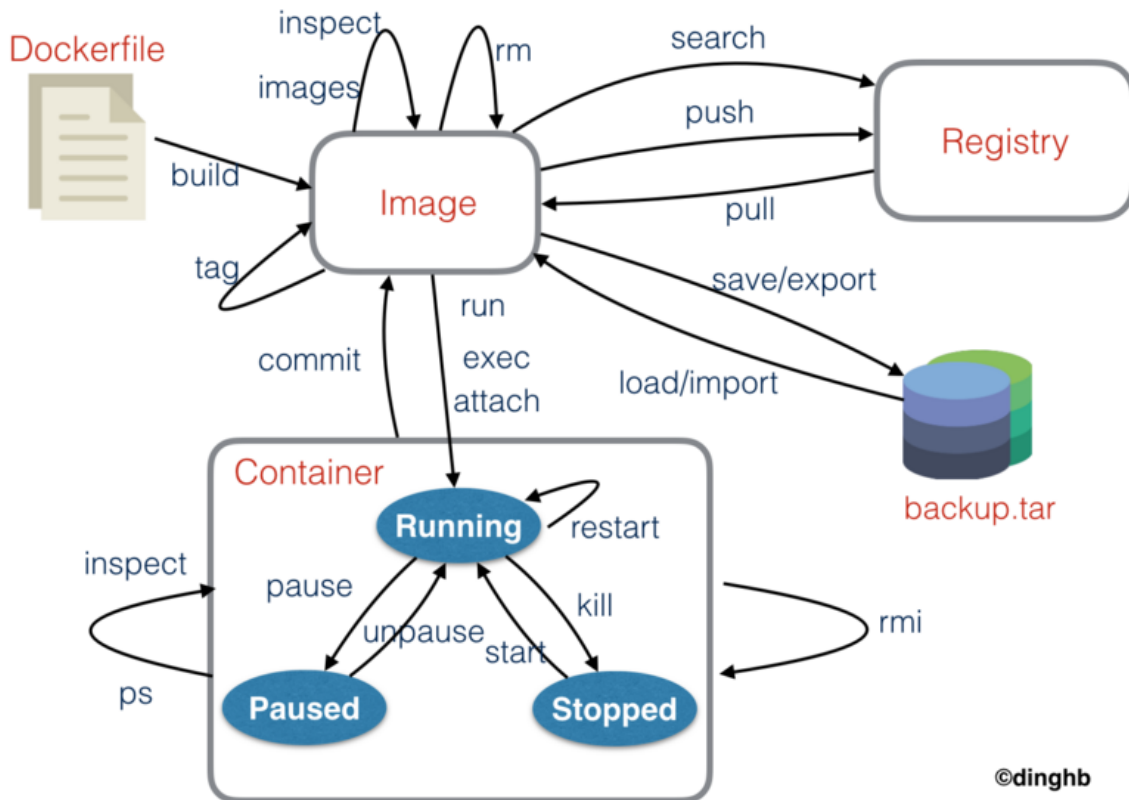
参考文档：<https://docs.docker.com/engine/reference/builder/>

容器级联

Compose是用于定义和运行多容器Docker应用程序的工具。通过Compose，可以使用YAML文件来配置应用程序的服务。

然后，使用一个命令，就可以从配置中创建并启动所有服务。

参考文档：<https://docs.docker.com/compose/compose-file/>



docker安装

Windows安装：

双击运行 Docker for Windows Installer.exe

Linux安装：

1、升级内核：

```
rpm -Uvh http://www.elrepo.org/elrepo-release-7.0-3.el7.elrepo.noarch.rpm
yum --enablerepo=elrepo-kernel install -y kernel-lt
grub2-set-default 'CentOS Linux (4.4.194-1.el7.elrepo.x86_64) 7 (Core)'
```

2、安装docker：

• 安装依赖

```
yum install -y yum-utils device-mapper-persistent-data lvm2
```

• 添加安装源

```
yum-config-manager --add-repo http://mirrors.aliyun.com/docker-
ce/linux/centos/docker-ce.repo
```

• 安装最新的docker

```
yum update -y && yum install -y docker-ce
```

• 设置daemon设置

```
mkdir /etc/docker
cat > /etc/docker/daemon.json <<EOF
```

```
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2",
  "storage-opts": [
    "overlay2.override_kernel_check=true"
  ]
}
EOF
```

- docker服务端重载config

```
systemctl daemon-reload
```

- 重启docker

```
systemctl restart docker
```

- 将docker加入到开机自启动

```
systemctl enable docker.service
```

3、安装docker-compose(liunx安装)

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.24.1/docker-
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```