



链滴

Docker 相关总结

作者: [EuthL](#)

原文链接: <https://ld246.com/article/1575293866522>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

docker有关总结

使用服务器经常用到docker部署，在这里简单总结一下我所接触到的docker,当然也为了方便自己以复习什么的

什么是docker?

Docker是一个开放源代码软件项目，让应用程序部署在软件货柜下的工作可以自动化进行，借此在Linux操作系统上，提供一个额外的软件抽象层，以及操作系统层虚拟化的自动管理机制。

这是从维基百科中查到的，按我的理解就是在服务器上提供了一个虚拟的空间可以方便软件运行、更及管理。

PS:一篇超级棒的说明docker和与docker相关的概念和技术文章：[可能是把Docker的概念讲的最清的一篇文章](#)

怎么使用?

一、在云服务器上安装docker

前提:

服务器系统：阿里云CentOS 7

1.添加yum源

```
# yum install epel-release -y
# yum clean all
# yum list
```

2.安装并运行Docker

```
# yum install docker-io -y
# systemctl start docker
```

3.检查安装结果

```
# docker info
```

二、docker基本概念

镜像

类似的，操作系统分为内核和用户空间。对于Linux而言，内核启动后，会挂载root文件系统为其供用户空间支持。而Docker镜像（Image），就相当于是一个root文件系统。

Docker镜像是一个特殊的文件系统，除了提供容器运行时所需的程序、库、资源、配置等文件外，包含了一些为运行时准备的一些配置参数（如匿名卷、环境变量、用户等）。镜像不包含任何动态数据，其内容在构建之后也不会被改变。

容器

镜像（Image）和容器（Container）的关系，就像是面向对象程序设计中的类和实例一样，镜像

静态的定义，容器是镜像运行时的实体。容器可以被创建、启动、停止、删除、暂停等。容器存储层生存周期和容器一样，容器消亡时，容器存储层也随之消亡。因此，任何保存于容器存储层的信息都随容器删除而丢失。

仓库

镜像构建完成后，可以很容易的在当前宿主上运行，但是，如果需要在其它服务器上使用这个镜像，我们就需要一个集中的存储、分发镜像的服务，Docker Registry就是这样的服务。

一个 Docker Registry中可以包含多个仓库 (Repository)；每个仓库可以包含多个标签 (Tag)；一个标签对应一个镜像。所以说：镜像仓库是Docker用来集中存放镜像文件的地方类似于我们之前常的代码仓库。

通常，一个仓库会包含同一个软件不同版本的镜像，而标签就常用于对应该软件的各个版本。我们可以通过 <仓库名>:<标签> 的格式来指定具体是这个软件哪个版本的镜像。如果不给出标签，将以 latest 作为默认标签。

参考文章： [可能是把Docker的概念讲的最清楚的一篇文章](#)

三、docker常用命令

操作容器

列出当前所有正在运行的容器

```
$docker ps
```

列出所有的容器

```
$docker ps -a
```

删除单个容器

```
$docker rm CONTAINER ID
```

删除所有容器

```
$docker rm $(docker ps -a -q)
```

停止、启动、杀死、重启一个容器

```
$docker stop CONTAINER ID
```

```
$docker start CONTAINER ID
```

```
$docker kill CONTAINER ID
```

```
$docker restart CONTAINER ID
```

操作镜像

列出镜像

```
$docker images
```

下载镜像

```
$docker pull image_name
```

删除一个或者多个镜像

```
$docker rmi image_name
```

另外：在删除时应该先停止容器，再删除这个容器或镜像

以上是我个人在接触docker中用的最多的几个命令，在此记录一下，方便以后添加和查找

后话

docker学习成本很低，也很使用，在这里我没有写的是关于docker启动镜像、网桥等操作主要是因在自己接触过程中还接触到了docker编排工具，感觉解决容器间通信等问题更方便，之后再开坑记录下吧。