

还在百度Docker命令？推荐一套我用起来特顺手的命令！

原创 梦想de星空 macrozheng 2020-11-11 09:02

收录于合集

#mall学习教程（参考篇）

27个

平时经常使用Docker来搭建各种环境，简单又好用！但是有时候往往会忘记命令，总结了一套非常实用的Docker命令，对于Java开发来说基本上够用了，希望对大家有所帮助！

Docker简介

Docker是一个开源的应用容器引擎，让开发者可以打包应用及依赖包到一个可移植的镜像中，然后发布到任何流行的Linux或Windows机器上。使用Docker可以更方便地打包、测试以及部署应用程序。

Docker环境安装

- 安装 `yum-utils` ；

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

- 为yum源添加docker仓库位置；

```
yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
```

- 安装docker服务；

```
yum install docker-ce
```

- 启动docker服务。

```
systemctl start docker
```

Docker镜像常用命令

搜索镜像

```
docker search java
```



下载镜像

```
docker pull java:8
```

查看镜像版本

由于 `docker search` 命令只能查找出是否有该镜像，不能找到该镜像支持的版本，所以我们需要通过 `Docker Hub` 来搜索支持的版本。

- 进入 `Docker Hub` 的官网，地址：<https://hub.docker.com>
- 然后搜索需要的镜像：



- 查看镜像支持的版本：



- 进行镜像的下载操作：

```
docker pull nginx:1.17.0
```

列出镜像

```
docker images
```



删除镜像

- 指定名称删除镜像：

```
docker rmi java:8
```

- 指定名称删除镜像（强制）：

```
docker rmi -f java:8
```

- 删除所有没有引用的镜像:

```
docker rmi $(docker images | grep none | awk '{print $3}')
```

- 强制删除所有镜像:

```
docker rmi -f $(docker images)
```

打包镜像

```
# -t 表示指定镜像仓库名称/镜像名称:镜像标签 .表示使用当前目录下的Dockerfile文件
docker build -t mall/mall-admin:1.0-SNAPSHOT .
```

Docker容器常用命令

新建并启动容器

```
docker run -p 80:80 --name nginx \
-e TZ="Asia/Shanghai" \
-v /mydata/nginx/html:/usr/share/nginx/html \
-d nginx:1.17.0
```

- **-p**: 将宿主机和容器端口进行映射, 格式为: 宿主机端口:容器端口;
- **--name**: 指定容器名称, 之后可以通过容器名称来操作容器;
- **-e**: 设置容器的环境变量, 这里设置的是时区;
- **-v**: 将宿主机上的文件挂载到宿主机上, 格式为: 宿主机文件目录:容器文件目录;
- **-d**: 表示容器以后台方式运行。

列出容器

- 列出运行中的容器:

```
docker ps
```



- 列出所有容器:

```
docker ps -a
```



停止容器

注意: `$ContainerName` 表示容器名称, `$ContainerId` 表示容器ID, 可以使用容器名称的命令, 基本也支持使用容器ID, 比如下面的停止容器命令。

```
docker stop $ContainerName(or $ContainerId)
```

例如:

```
docker stop nginx  
#或者  
docker stop c5f5d5125587
```

强制停止容器

```
docker kill $ContainerName
```

启动容器

```
docker start $ContainerName
```

进入容器

- 先查询出容器的 `pid` :

```
docker inspect --format "{{.State.Pid}}" $ContainerName
```

- 根据容器的`pid`进入容器:

```
nsenter --target "$pid" --mount --uts --ipc --net --pid
```



删除容器

- 删除指定容器:

```
docker rm $ContainerName
```

- 按名称通配符删除容器, 比如删除以名称 `mall-` 开头的容器:

```
docker rm `docker ps -a | grep mall-* | awk '{print $1}'`
```

- 强制删除所有容器;

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

查看容器的日志

- 查看容器产生的全部日志:

```
docker logs $ContainerName
```



- 动态查看容器产生的日志:

```
docker logs -f $ContainerName
```

查看容器的IP地址

```
docker inspect --format '{{ NetworkSettings.IPAddress }}' $ContainerName
```

```
docker inspect --format '{{ .NetworkSettings.IPAddress }}' $ContainerName
```



修改容器的启动方式

```
# 将容器启动方式改为always  
docker container update --restart=always $ContainerName
```

同步宿主机时间到容器

```
docker cp /etc/localtime $ContainerName:/etc/
```

指定容器时区

```
docker run -p 80:80 --name nginx \  
-e TZ="Asia/Shanghai" \  
-d nginx:1.17.0
```

查看容器资源占用状况

- 查看指定容器资源占用状况，比如cpu、内存、网络、io状态：

```
docker stats $ContainerName
```



- 查看所有容器资源占用情况：

```
docker stats -a
```



查看容器磁盘使用情况

```
docker system df
```



执行容器内部命令

```
docker exec -it $ContainerName /bin/bash
```



指定账号进入容器内部

```
# 使用root账号进入容器内部
docker exec -it --user root $ContainerName /bin/bash
```

查看所有网络

```
docker network ls
```

```
[root@local-linux ~]# docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
59b309a5c12f	bridge	bridge	local
ef34fe69992b	host	host	local
a65be030c632	none		

创建外部网络

```
docker network create -d bridge my-bridge-network
```


指定容器网络

```
docker run -p 80:80 --name nginx \  
--network my-bridge-network \  
-d nginx:1.17.0
```

修改镜像的存放位置

- 查看Docker镜像的存放位置：

```
docker info | grep "Docker Root Dir"
```



- 关闭Docker服务：

```
systemctl stop docker
```

- 先将原镜像目录移动到目标目录：

```
mv /var/lib/docker /mydata/docker
```

- 建立软连接：

```
ln -s /mydata/docker /var/lib/docker
```



- 再次查看可以发现镜像存放位置已经更改。



推荐阅读

- [mall-swarm微服务电商系统如何部署？用Jenkins自动化部署是真的香！](#)
- [看看人家那后端API接口写得，那叫一个优雅！](#)
- [MySQL如何实时同步数据到ES？试试这款阿里开源的神器！](#)
- [为何程序员上班时间总戴个耳机，看完恍然大悟...](#)

- [卸载Notepad++！事实已证明，它更好用](#)
- [还在手动整合Swagger？Swagger官方Starter是真的香！](#)
- [Nginx如何支持HTTPS？手把手教贼简单！](#)
- [搞定Mall项目中的权限管理功能，看懂这些问题就妥了！](#)
- [40K+Star！Mall电商实战项目开源回忆录！](#)
- [mall-swarm 微服务电商项目发布重大更新，打造Spring Cloud最佳实践！](#)



欢迎关注，点个在看

收录于合集 #mall学习教程（参考篇） 27

上一篇

Linux防火墙Firewall和Iptables的使用

下一篇

使用Maven插件构建Docker镜像

阅读原文

喜欢此内容的人还喜欢

项目中到底该不该用Lombok？

macrozheng

