



Docker虚拟化容器应用与实战

第2章 实战学习Docker常用命令

本章目标

- 掌握Docker的帮助命令
- 掌握Docker的镜像命令
- 掌握Docker的容器命令
- 掌握Docker的常用其他命令



帮助命令

- docker version
#显示docker的版本信息。
 - docker info
#显示docker的系统信息，包括镜像和容器的数量
 - docker 命令 --help
#帮助命令
- #帮助文档的地址：**
<https://docs.docker.com/engine/reference/commandline/build/>



镜像命令

- docker search
#搜索镜像

- docker pull
#下载镜像

- docker rmi
#删除镜像

docker rmi -f 镜像id
#强制删除指定的镜像

- docker rmi -f \$(docker images -aq)
#强制删除全部的镜像



- docker images
#查看所有本地主机上的镜像 可以使用docker image ls代替
- -a, --all Show all images (default hides intermediate images)
#列出 所有镜像
-q, --quiet Only show numeric IDs
只显示镜像的id
- docker images -aq
显示所有镜像的id



- `docker run 镜像id`
#新建容器并启动
- `--name="Name"` 容器名字 tomcat01 tomcat02 用来区分容器
- `-d` 后台方式运行
- `-it` 使用交互方式运行，进入容器查看内容
- `-p` 指定容器的端口
- `-p 8080(宿主机):8080(容器)`
- `-p ip:主机端口:容器端口`
- `-p 主机端口:容器端口(常用)`
- `-p 容器端口 容器端口`
- `-P(大写)` 随机指定端口



查看/删除容器

- docker ps
#列出所有运行的容器
- docker rm 容器id
#删除指定容器



测试、启动并进入容器

- `docker run -it centos /bin/bash`
- `[root@95039813da8d /]# ls`
`bin dev etc home lib lib64 lost+found media mnt opt proc root run sbin srv sys`
`tmp usr var`
- `[root@95039813da8d /]# exit`
#直接退出容器
- `[root@95039813da8d /] ctrl +P +Q`
#容器不停止退出



列出所有运行的容器

- docker ps命令
#列出当前正在运行的容器
- -a, --all 显示所有的容器，包括未运行的
- -q, --quiet 静默模式，只显示容器编号
- docker ps -aq



退出容器

- exit
#容器直接退出
- ctrl +P +Q
#容器不停止退出



启动和停止容器操作

- docker start 容器id
#启动容器
- docker restart 容器id
#重启容器
- docker stop 容器id
#停止当前正在运行的容器
- docker kill 容器id
#强制停止当前容器



后台启动命令

- `docker run -d centos`

命令 `docker run -d` 镜像名

- `docker ps`

问题 `docker ps`. 发现 `centos` 停止了

`docker` 容器使用后台运行，就必须要有要一个前台进程，`docker` 发现没有应用，就会自动停止，容器启动后，发现自己没有提供服务，就会立刻停止，就是没有程序了



赛博梦工厂

Cyber Works

1、下载centos镜像并运行

- docker pull centos
- docker run -it --name centos01 centos /bin/bash

2、下载tomcat镜像并运行

- docker pull tomcat:9
- docker run -d --name tomcat01 tomcat:9



- `docker logs --help`
- * `-f`: 跟踪日志输出
- `--since`: 显示某个开始时间的所有日志
- * `--tail`: 仅列出最新第N条容器日志
- * `-t`: 显示时间戳



查看容器中进程信息 ps

- docker top 容器id



查看镜像的元数据

- docker inspect 容器id

#测试

```
docker inspect 321bcae33d [ { "Id":  
  "321bcae33d15da8280bcac1d2bc1141d213bcc8f8e792edfd832ff61ae5066",  
  "Created": "2020-05-15T05:22:05.515909071Z", "Path": "/bin/sh", ... } ]
```



进入当前正在运行的容器

我们通常容器都是使用后台方式运行的，需要进入容器，修改一些配置

- `docker exec -it 容器id bashshell`

- `docker ps`
#测试

- `docker exec -it 321bcae33d /bin/bash`

- `[root@321bcae33d /]#`



从容器内拷贝到主机上

- `docker cp 容器id:容器内路径 宿主机目的路径`

#进入docker容器内部

- `docker exec -it 321bcae33d /bin/bash`

- `[root@321bcae33d /]# ls`

`bin etc lib lost+found mnt proc run srv tmp var dev home lib64 media opt root sbin
sys usr`



赛博梦工厂

Cyber Works

小结

- 掌握Docker的帮助命令
- 掌握Docker的镜像命令
- 掌握Docker的容器命令
- 掌握Docker的常用其他命令





谢谢观赏