**Docker**

# docker安装

## 1.1 安装telnet

# docker

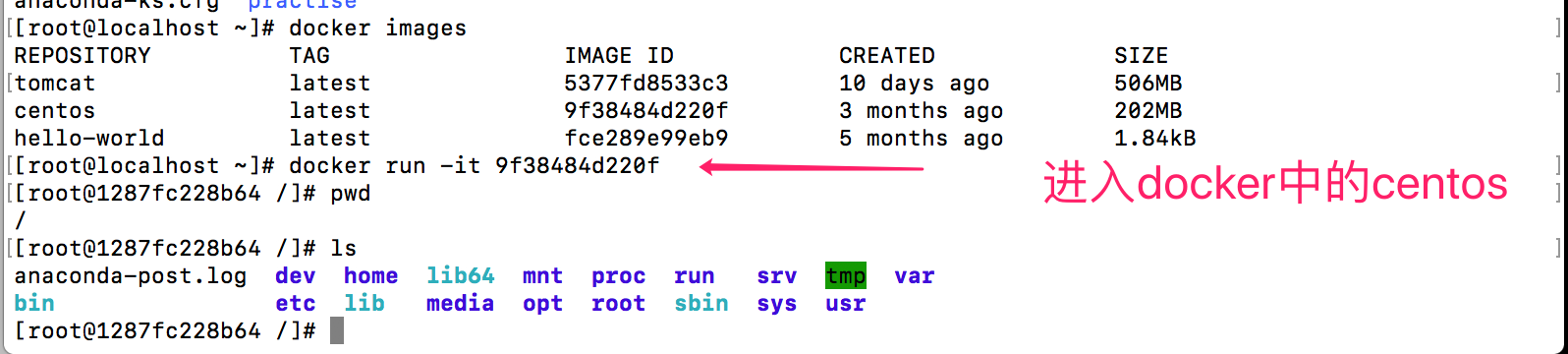
# docker容器命令

## docker pull

## 新建并启动容器

docker run [options] image [command] [arg…]

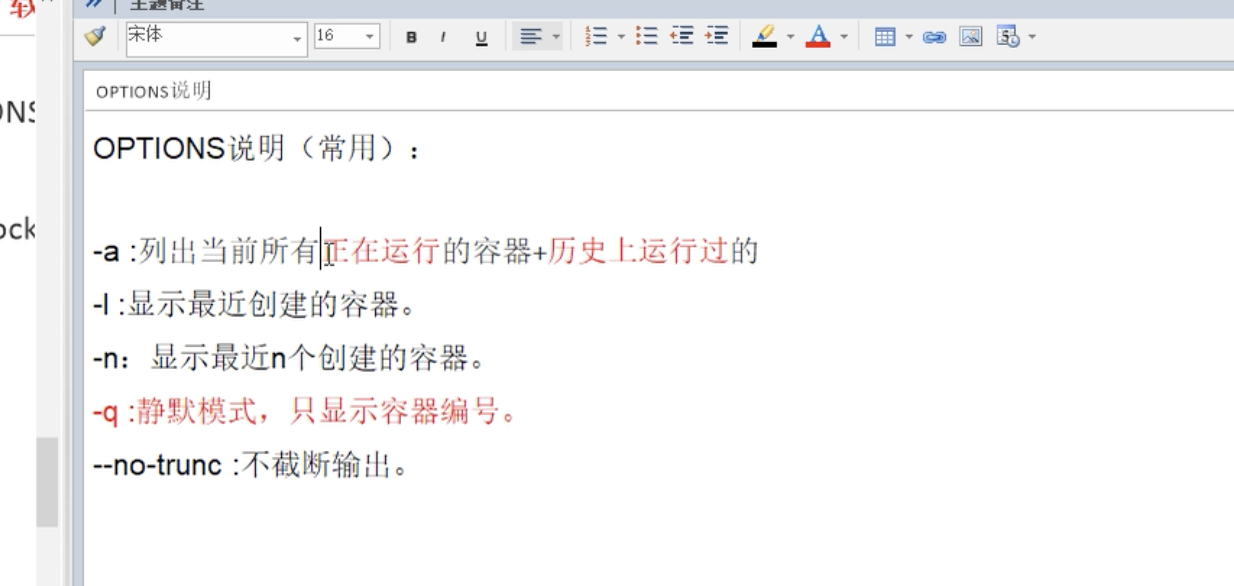




## 列出当前所有正在运行的容器

docker ps [options]





## 退出容器

### 4.4.1 exit容器停止退出

### 4.4.2 ctrl+P+Q（容器不停止退出）

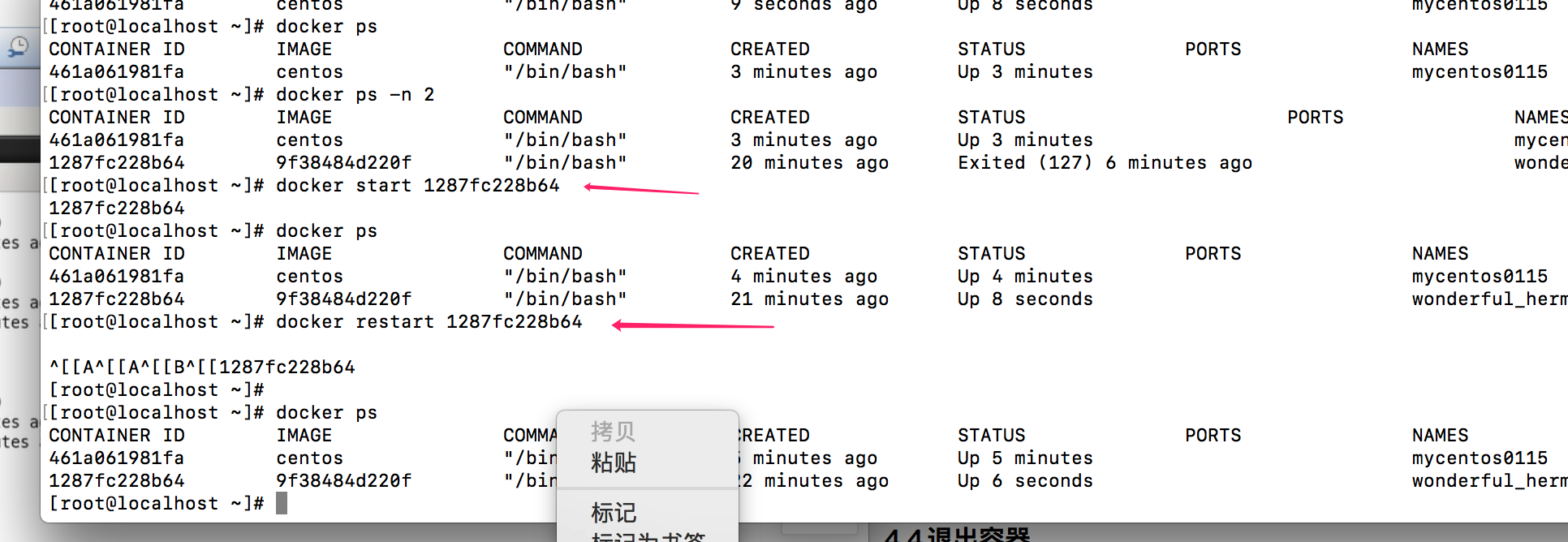
## 启动，重启，停止

docker start 容器id

docker restart 容器id

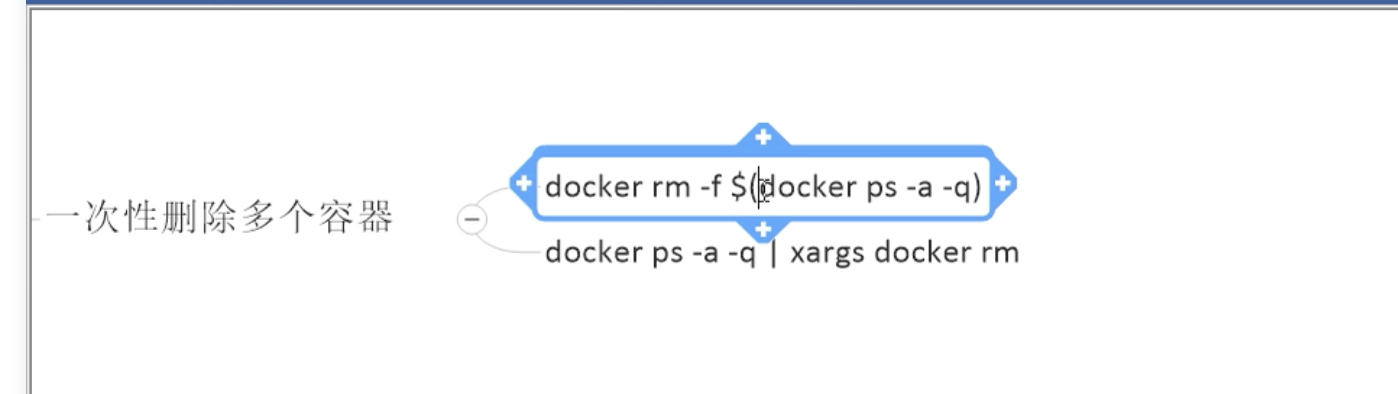
docker stop 容器id 缓慢停止

docker kill 容器id 强制停止



## 删除已停止的容器

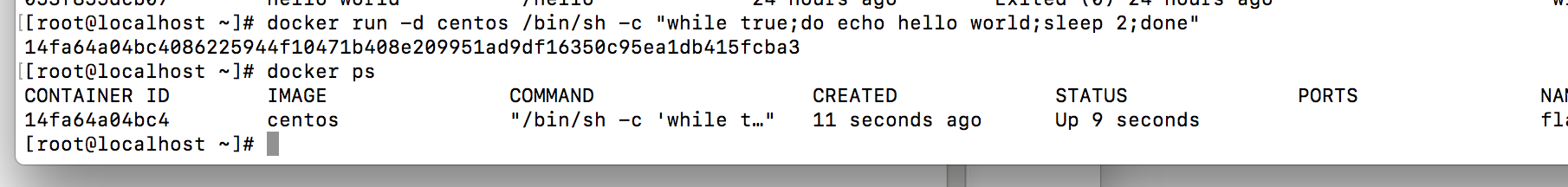
docker rmi

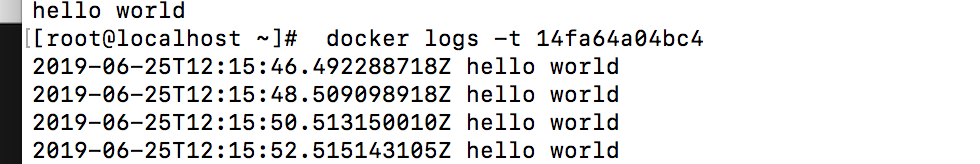


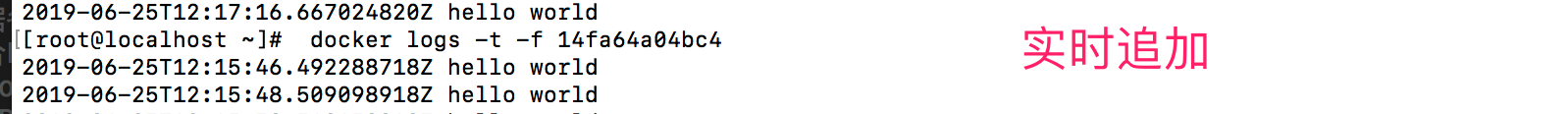
## 删启动守护式容器docker run –d 容器名

## 查看容器日志

docker logs –f –t –tail 容器id

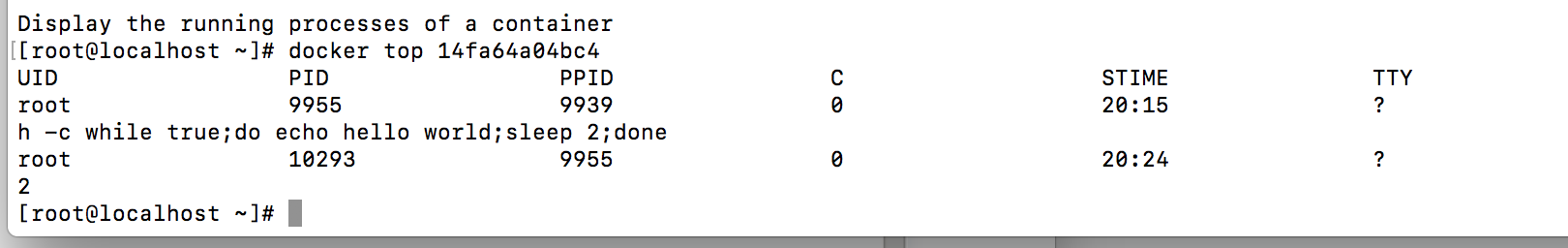






## 查看容器内进程

docker top



## 查看容器内部细节

docker inspect 容器id

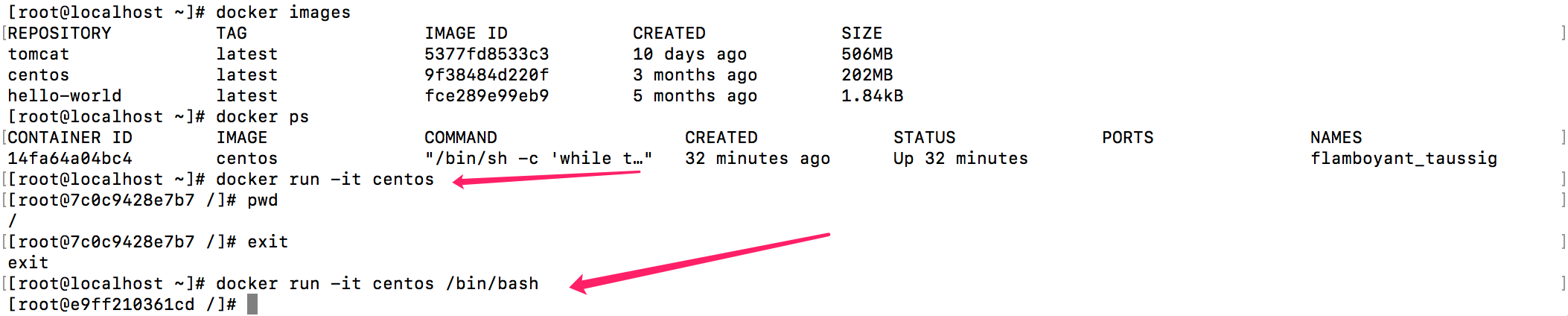
## 进入正在运行的容器内部进行交互

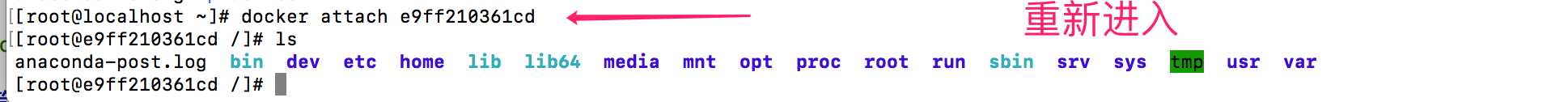
docker exec –it 容器id bashShell

重新进入：docker attach 容器Id

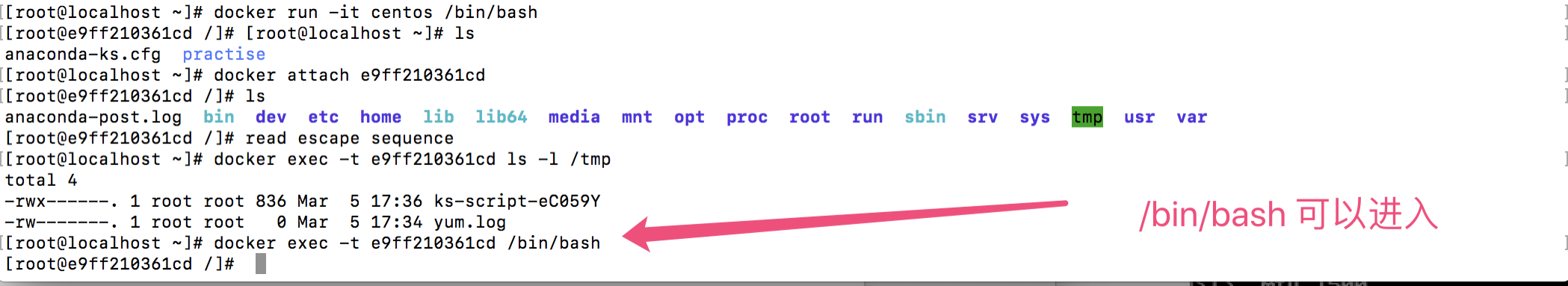
区别：attach直接进入容器启动命令的终端，不会启动新的进程

exec：是在容器中打开的新终端，并且可以启动新的进程









## 从容器内拷贝文件到主机上

docker cp 容器id：容器内路径 目的主机路径



## 小结

# docker镜像

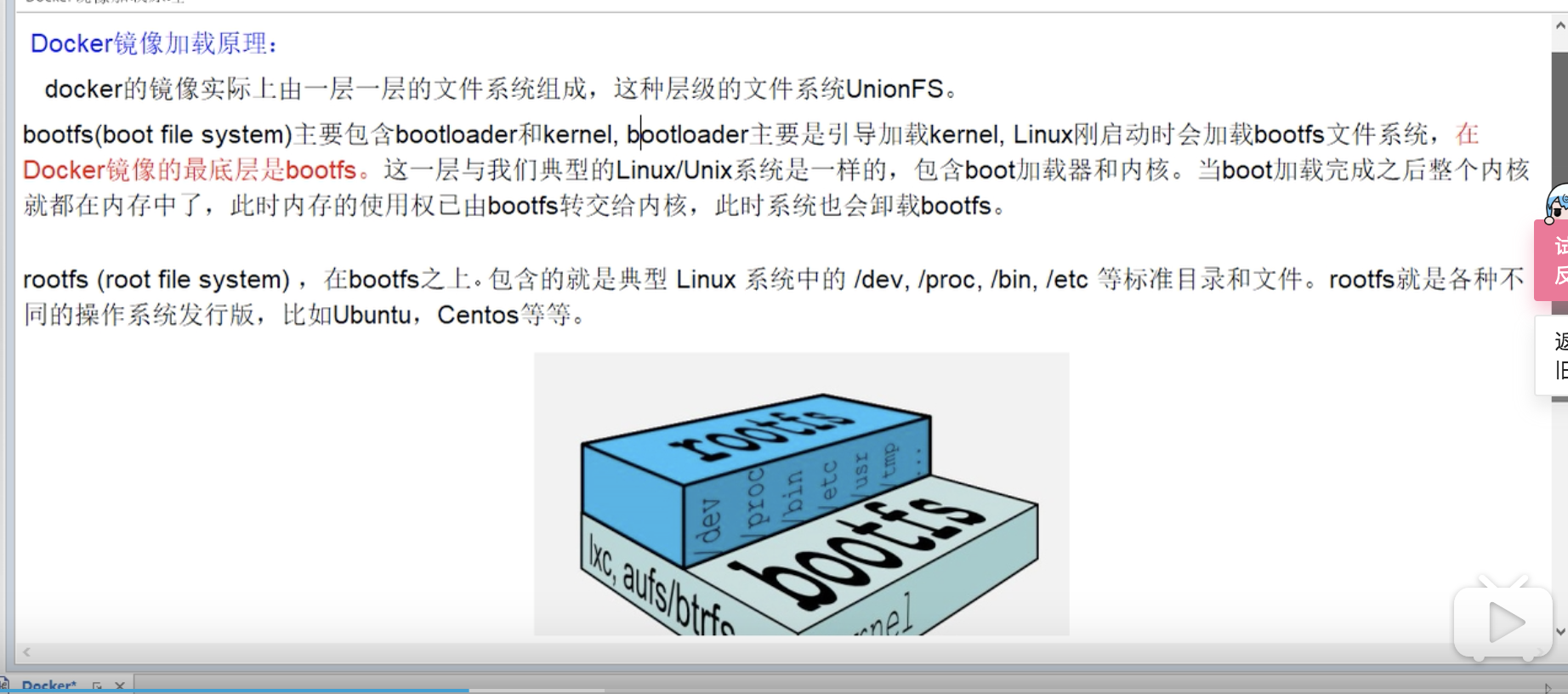
## 是什么

轻量级、可执行的独立软件包，用来打包软件运行环境和基于运行环境开发的软件，它是包含运行某个软件所需要的所有内容，包括代码、运行时、库、环境变量和配置文件

### UnionFS联合文件系统

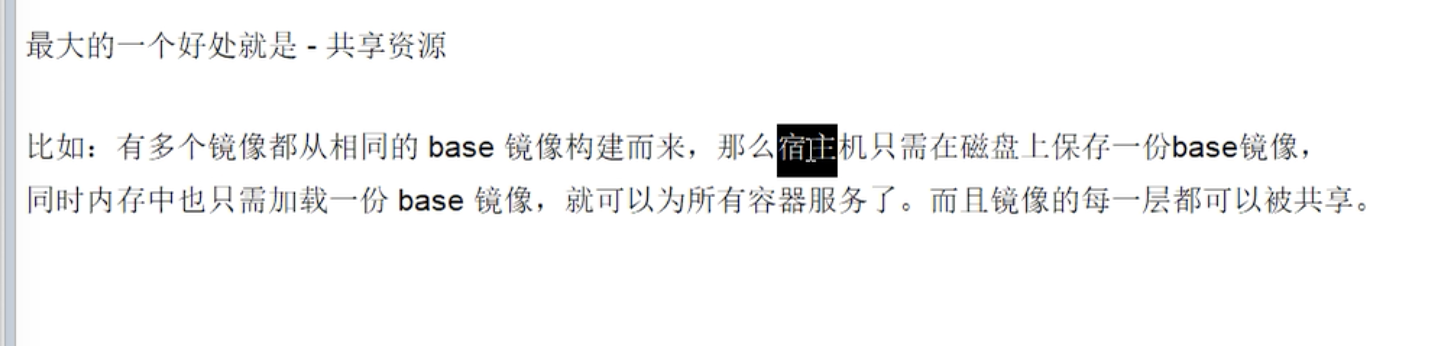


### Docker加载原理



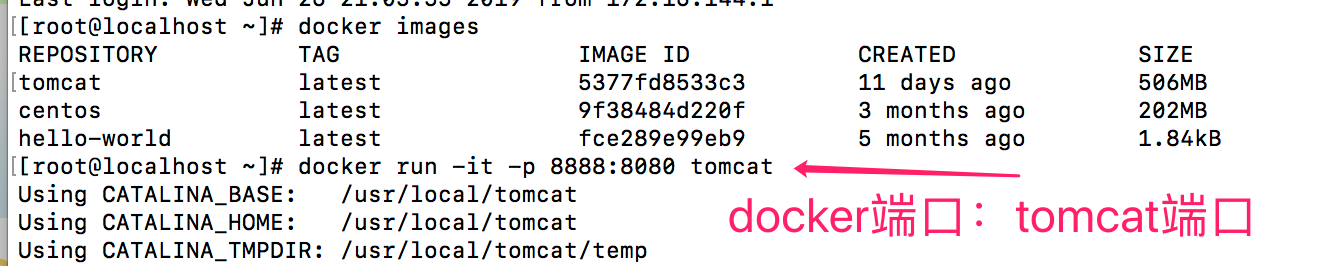
### Docker分层的镜像

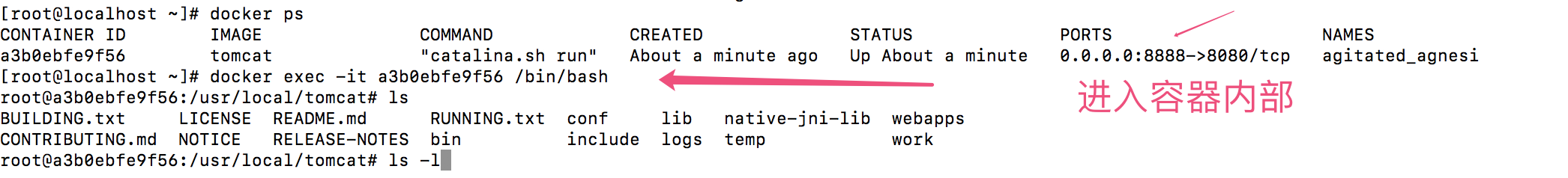
### Docker为什么采用分层镜像



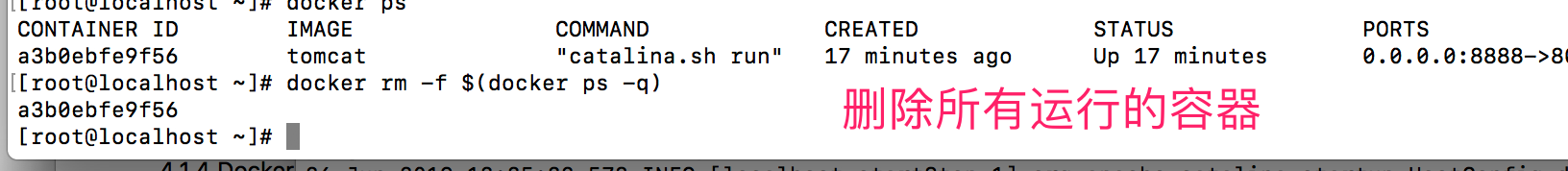
## 镜像命令commit

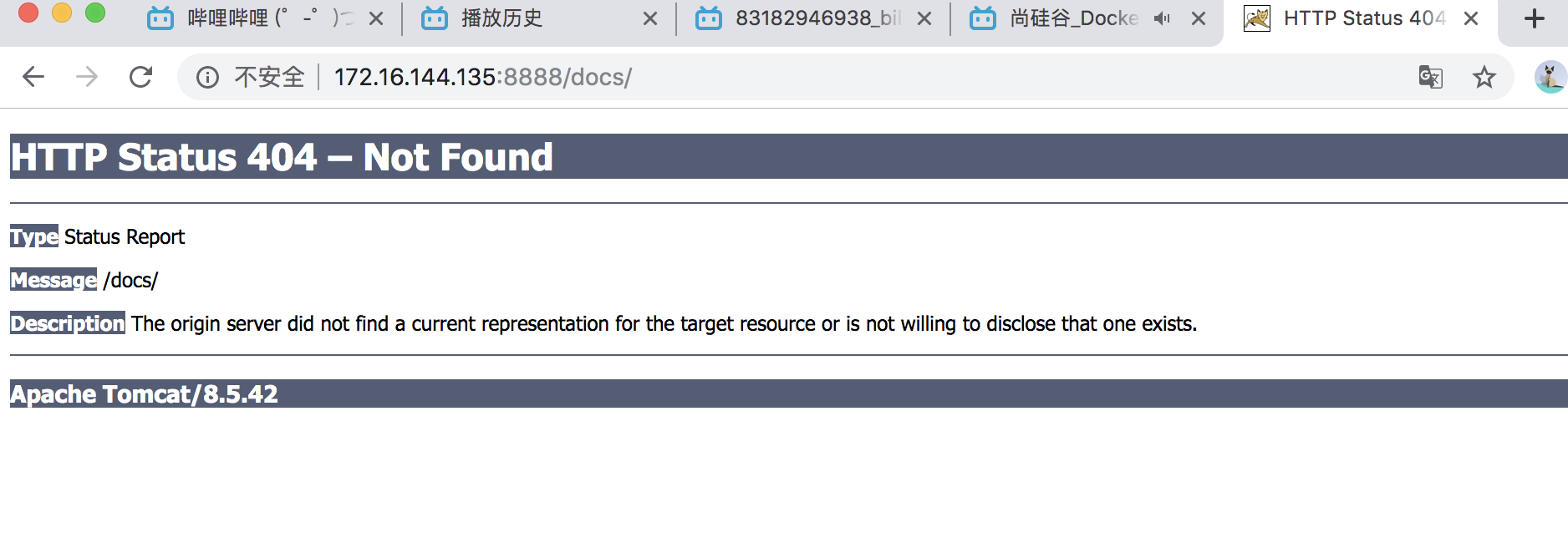
### docker commit提交容器副本使之成为一个新的镜像

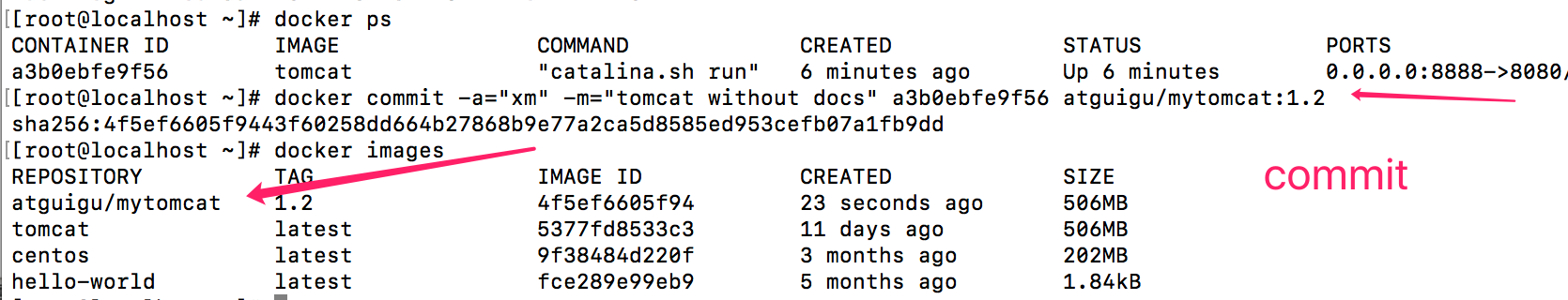












# docker容器数据卷

## 是什么

## 能干嘛



## 数据卷

## 数据卷容器