

VMware->docker->k8s

硬件->操作系统->应用(qq,wechat等等) 目前的架构

多个硬件->hypervisor(虚拟化操作系统-可以把底层的多个硬件变成一个大的硬件，虚拟化技术)->操作系统->应用

docker可以使用宿主机的全部资源，虚拟机只能用分配给他的资源

docker引擎包含：

Docker守护进程<-rest api<-docker命令行工具 (container\image\network\data volumes)

Docker image(class) docker container(instance)

docker build\docker pull\docker run

apt

Yum 1 curl -fsSL <https://get.docker.com> -o get-docker.sh 2 sudo sh get-docker.sh --mirror Aliyun
使用阿里云镜像安装DOCKER 3 在/etc/DOCKER目录创建daemon.json文件，添加如下内容 {
"registry-mirrors": [<https://alzgoonw.mirror.aliyuncs.com>] } 4 执行命令
systemctl daemon-reload systemctl restart docker 5 使用docker docker pull tomcat[:tag] docker
images docker rmi <image_id> docker run -it --rm -p 8888:8080 tomcat:tag docker ps <-a> 如果没有
-a表示查看当前运行的容器，如果有-a表示所有的容器 docker rm <容器id> docker run -it --rm \n
ubuntu:16.04 \ bash

1 docker image rm和docker rmi都是删除镜像 docker rm是删除容器 2 cp rm mv 3 echo > >> 4
docker rm id1 id2 id3 可以删除多个，多个ID之间通过空格分割 5 docker exec -it <dockerid>-docker
ps出来的结果> bash 交互的方式进入容器，进入某个已经启动的容器 docker run -it tomcat bash 每次
都是新启动一个容器 docker run -p 8080:8080 tomcat 启动容器，前一个是宿主机端口，后一个是容器
端口 6 vi Dockerfile FROM tomcat <> 必填 []选填 RUN cd /usr/local/tomcat/webapps 和 WORKDIR
/usr/local/tomcat/webapps的区别 RUN mkdir ROOT RUN cd ROOT RUN touch index.html RUN
echo "hello my docker"> /usr/local/tomcat/webapps/ROOT/index.html 7 docker build -t mydocker
. 构建一个自己的镜像，名字是mydocker 使用当前目录的Dockerfile构建 8 docker image prune 删除
没有容器的虚悬镜像 先删除无用的容器 docker ps -a docker rm id2和id1之间使用空格分开 9
Dockerfile中的所有操作都是在docker引擎中的操作，不是在本地的操作，如果要操作本地文件的话，
必须把本地文件放在Dockerfile所在目录，使用docker build命令把dockerfile所在目录整体打包给
docker引擎所在服务器，同时dockerfile中的命令都必须指定工作目录，使用WORKDIR,不能使用CD命
令。 10 COPY复制文件 ADD更高级的复制文件 CMD容器启动命令 ENTRYPOINT入口点 ENV设置环境变量
VOLUME定义匿名卷 EXPOSE暴露端口 WORKDIR指定工作目录 RUN每RUN一次相当于启动一个容器
FROM dockerfile必须是FROM开始 11 docker run -p 8080:8080 --name mytomcat -d tomcat --name
重新命名 -d 守护态运行 docker run -p 8081:8080 --name secondtomcat -d tomcat docker restart
tomcat docker stop tomcat docker container prune docker rm 12 docker run -p 8080:8080 --name
mytomcat -d -v /usr/local/docker/tomcat/ROOT:/usr/local/tomcat/webapps/Root tomcat 13
docker run -p 3306:3306 --name mysql -v /usr/local/docker/mysql/conf:/etc/mysql -v
/usr/local/docker/mysql/logs:/var/log/mysql -v /usr/local/docker/mysql/data:/var/lib/mysql -e
MYSQL_ROOT_PASSWORD=123456 -d mysql 14 whereis mysql 15 docker cp mysql:/etc/mysql . 拷贝
容器中的文件到宿主机中 16 docker logs tomcat 查看日志文件 docker logs -f tomcat 17 安装docker-
compose curl -L <https://get.daocloud.io/docker/compose/releases/download/1.26.0/docker-comp>
[ose](https://get.daocloud.io/docker/compose/releases/download/1.26.0/docker-compose) -u name -s -u name -m > /usr/local/bin/docker-compose sudo chmod +x /usr/local/bin/docker-
compose docker-compose up -d docker-compose down 必须在docker-compose.yml所在目录执行
docker-compose logs tomcat

