Docker：它是一个开源的软件项目，在Linux操作系统上，docker提供了一个额外的软件抽象层及操作系统层虚拟化的自动管理机制。

物理机：

1. 安装系统
2. 依赖环境
   1. Java – jdk jre
   2. NodeJS – Node
   3. PHP – PHP
3. 应用程序
4. 加一个物理机—> 提高并发量

虚拟机：

KVM Xen

1. 把一个物理机虚拟机虚拟成多个机器
2. 把依赖环境打成一个系统的模板

容器化：

Docker

1. 镜像基础
   1. 依赖环境的镜像
      1. Java – Java基础的基础镜像
      2. PHP – PHP基础的基础镜像
   2. 根据基础镜像 – 放入自己的代码或者包
      1. 生产一个新镜像
      2. 程序镜像
   3. 镜像 – 按层存储
      1. A – Java 🡪 JDK1.8
         1. a.jar
      2. B – Java -> JDK1.8
         1. b.jar
2. 启动时间特别，秒级启动

容器：把自己的应用程序，根据某个依赖的基础镜像，生成一个应用程序镜像

应用程序镜像，可以运行在任何部署了Docker环境的机器上。

Docker基本命令

# 查看Docker版本

[root@k8s-master01 ~]# docker version

Client: Docker Engine - Community

Version: 19.03.4

API version: 1.40

Go version: go1.12.10

Git commit: 9013bf583a

Built: Fri Oct 18 15:52:22 2019

OS/Arch: linux/amd64

Experimental: false

Server: Docker Engine - Community

Engine:

Version: 19.03.4

API version: 1.40 (minimum version 1.12)

Go version: go1.12.10

Git commit: 9013bf583a

Built: Fri Oct 18 15:50:54 2019

OS/Arch: linux/amd64

Experimental: false

containerd:

Version: 1.2.6

GitCommit: 894b81a4b802e4eb2a91d1ce216b8817763c29fb

runc:

Version: 1.0.0-rc8

GitCommit: 425e105d5a03fabd737a126ad93d62a9eeede87f

docker-init:

Version: 0.18.0

GitCommit: fec3683

Docker详细信息

[root@k8s-master01 ~]# docker info

Client:

Debug Mode: false

Server:

Containers: 8

Running: 4

Paused: 0

Stopped: 4

Images: 8

Server Version: 19.03.4

Storage Driver: overlay2

# aufs、overlay brtfs

Backing Filesystem: xfs

Supports d\_type: true

Native Overlay Diff: true

Logging Driver: json-file

# json-file: 存在本地

Cgroup Driver: cgroupfs

Plugins:

Volume: local

Network: bridge host ipvlan macvlan null overlay

Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog

Swarm: inactive

Runtimes: runc

Default Runtime: runc

Init Binary: docker-init

containerd version: 894b81a4b802e4eb2a91d1ce216b8817763c29fb

runc version: 425e105d5a03fabd737a126ad93d62a9eeede87f

init version: fec3683

Security Options:

seccomp

Profile: default

Kernel Version: 4.18.0-80.el8.x86\_64

Operating System: CentOS Linux 8 (Core)

OSType: linux

Architecture: x86\_64

CPUs: 2

Total Memory: 1.764GiB

Name: k8s-master01

ID: PRWL:PVRE:U7JQ:LNM6:SNLN:4QDV:URQA:MWQF:XXCE:VOT3:53GL:ECC6

Docker Root Dir: /var/lib/docker

# 可以更改的，ssd的硬盘。最后使用一个单独的磁盘进行挂载

Debug Mode: false

Registry: https://index.docker.io/v1/

Labels:

Experimental: false

Insecure Registries:

127.0.0.0/8

Live Restore Enabled: false

搜索镜像：

[root@k8s-master01 ~]# docker search centos

NAME DESCRIPTION STARS OFFICIAL AUTOMATED

centos The official build of CentOS. 6054 [OK]

ansible/centos7-ansible Ansible on Centos7 130 [OK]

consol/centos-xfce-vnc Centos container with "headless" VNC session… 116 [OK]

jdeathe/centos-ssh OpenSSH / Supervisor / EPEL/IUS/SCL Repos - … 114 [OK]

centos/systemd systemd enabled base container. 84 [OK]

centos/mysql-57-centos7 MySQL 5.7 SQL database server 77

imagine10255/centos6-lnmp-php56 centos6-lnmp-php56 58 [OK]

tutum/centos Simple CentOS docker image with SSH access 47

centos/postgresql-96-centos7 PostgreSQL is an advanced Object-Relational … 45

kinogmt/centos-ssh CentOS with SSH 29 [OK]

pivotaldata/centos-gpdb-dev CentOS image for GPDB development. Tag names… 12

guyton/centos6 From official centos6 container with full up… 10 [OK]

drecom/centos-ruby centos ruby 6 [OK]

centos/tools Docker image that has systems administration… 6 [OK]

pivotaldata/centos Base centos, freshened up a little with a Do… 4

pivotaldata/centos-gcc-toolchain CentOS with a toolchain, but unaffiliated wi… 3

pivotaldata/centos-mingw Using the mingw toolchain to cross-compile t… 3

darksheer/centos Base Centos Image -- Updated hourly 3 [OK]

miko2u/centos6 CentOS6 日本語環境 2 [OK]

blacklabelops/centos CentOS Base Image! Built and Updates Daily! 1 [OK]

mcnaughton/centos-base centos base image 1 [OK]

indigo/centos-maven Vanilla CentOS 7 with Oracle Java Developmen… 1 [OK]

pivotaldata/centos6.8-dev CentosOS 6.8 image for GPDB development 0

smartentry/centos centos with smartentry 0 [OK]

pivotaldata/centos7-dev CentosOS 7 image for GPDB development 0

[root@k8s-master01 ~]# docker search nginx

NAME DESCRIPTION STARS OFFICIAL AUTOMATED

nginx Official build of Nginx. 13358 [OK]

Docker拉取镜像：

拉取一个镜像到本地：

[root@k8s-master01 ~]# docker pull alpine:latest

latest: Pulling from library/alpine

df20fa9351a1: Already exists

Digest: sha256:185518070891758909c9f839cf4ca393ee977ac378609f700f60a771a2dfe321

Status: Downloaded newer image for alpine:latest

docker.io/library/alpine:latest

Docker run：启动一个镜像

1. 前台启动
   1. [root@k8s-master01 ~]# docker run -ti centos:8 bash
   2. [root@55eb31fec62e /]# whoami
   3. root
2. 后台启动

-d 后台启动一个镜像

查看容器日志：

CMD和ENTRYPIOINT 必须要有一个

CMD可以被覆盖，如果有ENTRYPIOINT的话，CMD就是ENTRYPIOINT的参数。

ENTRYPIOINT – 》 COMMAND

CMD –》 arg

FROM centos:8

LABEL maintainer="test dockerfile"

LABEL test=dockerfile

RUN useradd dot

RUN mkdir /opt/dot

CMD [ "sh", "-c", "echo 1"]

#RUN useradd dot && /opt/dot

FROM centos:8

LABEL maintainer="test dockerfile"

LABEL test=dockerfile

ENV test\_env1 env1

ENV test\_env2 env2

RUN useradd dot

RUN mkdir /opt/dot

#ENTRYPOINT ["echo"]

ENV env1=test1 env2=test2

ADD ./index.tar.gz /opt/

COPY ./index.tar.gz /opt/dot/

WORKDIR /opt/dot

USER 1000

CMD pwd ; ls

#RUN useradd dot && /opt/dot

制作小镜像：

一定不要使用centos镜像

Alpine，busybox，scratch，Debian

Glibc： node:slim python:slim net

使用多阶段构建：

编译操作和生成最终镜像的操作

# build step

FROM golang:1.14.4-alpine as builder

WORKDIR /opt

COPY main.go /opt

RUN go build /opt/main.go

CMD "./main"

# create real app image

FROM alpine:3.8

COPY --from=builder /opt/main /

CMD "./opt/main"

FROM php:7.1.22-fpm-alpine

RUN apk add --no-cache binutils freetype libpng libjpeg-turbo freetype-dev libpng-dev libjpeg-turbo-dev libc6-compat libxml2 libxml2-dev libmcrypt libmcrypt-dev libc-dev icu-dev gettext-dev openssl-dev bzip2-dev

RUN docker-php-ext-install pdo pdo\_mysql mcrypt zip gd pcntl opcache bcmath

#RUN docker-php-ext-install gettext

RUN docker-php-ext-install mysqli

#RUN apk add --no-cache php7-sysvsem php7-pdo\_dblib php7-sockets php-soap php7-xmlrpc

##RUN apk add --no-cache php7-sysvsem php7-pdo\_dblib php7-sockets php-soap php7-xmlrpc

##RUN apk add --no-cache freetds-dev

##RUN docker-php-ext-install pdo\_dblib

#RUN docker-php-ext-install soap

#RUN docker-php-ext-install sockets

#RUN docker-php-ext-install sysvsem

#RUN docker-php-ext-install xmlrpc

#RUN apk add --no-cache freetds-dev

#RUN docker-php-ext-install pdo\_dblib

#RUN docker-php-ext-configure gd --with-freetype-dir=/usr/include/ --with-jpeg-dir=/usr/include/

#RUN docker-php-ext-install -j$(nproc) gd

#FROM php:7.1.22-fpm-alpine

#COPY --from=0 /usr/local/lib/php/extensions/no-debug-non-zts-20160303 /usr/local/lib/php/extensions/no-debug-non-zts-20160303

#RUN apk add --no-cache freetds-dev php7-sysvsem php7-pdo\_dblib php7-sockets php-soap php7-xmlrpc binutils freetype libpng libjpeg-turbo freetype-dev libpng-dev libjpeg-turbo-dev libc6-compat libxml2 libxml2-dev libmcrypt libmcrypt-dev libc-dev icu-dev gettext-dev openssl-dev bzip2-dev && cd /usr/local/lib/php/extensions/no-debug-non-zts-20160303 && docker-php-ext-enable \*.so && rm -rf /var/cache/apk/\*