## 2.1 Docker基本概念

### 2.1.1 Docker基本命令

[root@k8s-master01 ~]# docker version

Client:

Version: 17.09.1-ce

API version: 1.32

Go version: go1.8.3

Git commit: 19e2cf6

Built: Thu Dec 7 22:23:40 2017

OS/Arch: linux/amd64

Server:

Version: 17.09.1-ce

API version: 1.32 (minimum version 1.12)

Go version: go1.8.3

Git commit: 19e2cf6

Built: Thu Dec 7 22:25:03 2017

OS/Arch: linux/amd64

Experimental: false

[root@k8s-master01 ~]# docker info

Containers: 22

Running: 21

Paused: 0

Stopped: 1

Images: 18

Server Version: 17.09.1-ce

Storage Driver: overlay2

Backing Filesystem: xfs

Supports d\_type: true

Native Overlay Diff: true

Logging Driver: json-file

Cgroup Driver: cgroupfs

Plugins:

Volume: local

Network: bridge host macvlan null overlay

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

Swarm: inactive

Runtimes: runc

Default Runtime: runc

Init Binary: docker-init

containerd version: 06b9cb35161009dcb7123345749fef02f7cea8e0

runc version: 3f2f8b84a77f73d38244dd690525642a72156c64

init version: 949e6fa

Security Options:

seccomp

Profile: default

Kernel Version: 4.18.9-1.el7.elrepo.x86\_64

Operating System: CentOS Linux 7 (Core)

OSType: linux

Architecture: x86\_64

CPUs: 4

Total Memory: 3.848GiB

Name: k8s-master01

ID: HM66:LH4K:PNES:GFJX:TKNX:TLOH:WONE:KLHT:YRB3:3KAR:3WZJ:HYOX

Docker Root Dir: /var/lib/docker

Debug Mode (client): false

Debug Mode (server): false

Username: dotbalo

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

Experimental: false

Insecure Registries:

127.0.0.0/8

Live Restore Enabled: false

[root@k8s-master01 docker]# ls

builder containers image network overlay2 plugins swarm tmp trust volumes

容器日志打印目录

/var/lib/docker/containers/f3970439b08a2f4d586d62f30b6ee1ef4bf87e3c6f6b75f7f0721fc0bf0f7d03

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

NAME DESCRIPTION STARS OFFICIAL AUTOMATED

nginx Official build of Nginx. 10749 [OK]

jwilder/nginx-proxy Automated Nginx reverse proxy for docker c... 1507 [OK]

richarvey/nginx-php-fpm Container running Nginx + PHP-FPM capable ... 675 [OK]

jrcs/letsencrypt-nginx-proxy-companion LetsEncrypt container to use with nginx as... 469 [OK]

webdevops/php-nginx Nginx with PHP-FPM 120 [OK]

kitematic/hello-world-nginx A light-weight nginx container that demons... 119

zabbix/zabbix-web-nginx-mysql Zabbix frontend based on Nginx web-server ... 86 [OK]

bitnami/nginx Bitnami nginx Docker Image 60 [OK]

linuxserver/nginx An Nginx container, brought to you by Linu... 51

1and1internet/ubuntu-16-nginx-php-phpmyadmin-mysql-5 ubuntu-16-nginx-php-phpmyadmin-mysql-5 48 [OK]

tobi312/rpi-nginx NGINX on Raspberry Pi / armhf 23 [OK]

nginx/nginx-ingress NGINX Ingress Controller for Kubernetes 15

blacklabelops/nginx Dockerized Nginx Reverse Proxy Server. 12 [OK]

wodby/drupal-nginx Nginx for Drupal container image 11 [OK]

centos/nginx-18-centos7 Platform for running nginx 1.8 or building... 10

nginxdemos/hello NGINX webserver that serves a simple page ... 9 [OK]

webdevops/nginx Nginx container 8 [OK]

centos/nginx-112-centos7 Platform for running nginx 1.12 or buildin... 6

1science/nginx Nginx Docker images that include Consul Te... 4 [OK]

travix/nginx NGinx reverse proxy 2 [OK]

mailu/nginx Mailu nginx frontend 2 [OK]

pebbletech/nginx-proxy nginx-proxy sets up a container running ng... 2 [OK]

toccoag/openshift-nginx Nginx reverse proxy for Nice running on sa... 1 [OK]

ansibleplaybookbundle/nginx-apb An APB to deploy NGINX 0 [OK]

wodby/nginx Generic nginx

Docker search搜索的是公网的镜像，地址：<https://hub.docker.com/>

# 把公网上的镜像拉取到本地服务器

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

Using default tag: latest

latest: Pulling from library/nginx

Digest: sha256:b543f6d0983fbc25b9874e22f4fe257a567111da96fd1d8f1b44315f1236398c

Status: Image is up to date for nginx:latest

拉取指定版本

[root@k8s-master01 ~]# docker pull nginx:1.15

1.15: Pulling from library/nginx

Digest: sha256:b543f6d0983fbc25b9874e22f4fe257a567111da96fd1d8f1b44315f1236398c

Status: Downloaded newer image for nginx:1.15

docker push：把本地的镜像推送到公网仓库中，或者是公司内部仓库中

[root@k8s-master01 ~]# docker images | grep nginx-v2

nginx-v2 latest 3d9c6e44d3db 3 hours ago 109MB

[root@k8s-master01 ~]# docker tag nginx-v2 dotbalo/nginx-v2:test

[root@k8s-master01 ~]# docker images | grep nginx-v2

dotbalo/nginx-v2 test 3d9c6e44d3db 3 hours ago 109MB

nginx-v2 latest 3d9c6e44d3db 3 hours ago 109MB

[root@k8s-master01 ~]# docker login

Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.

Username (dotbalo): dotbalo

Password:

Login Succeeded

[root@k8s-master01 ~]# docker push dotbalo/nginx-v2:test

The push refers to a repository [docker.io/dotbalo/nginx-v2]

2eaa7b5717a2: Mounted from dotbalo/nginx

a674e06ede38: Mounted from dotbalo/nginx

b7efe781401d: Mounted from dotbalo/nginx

c9c2a3696080: Mounted from dotbalo/nginx

7b4e562e58dc: Mounted from dotbalo/nginx

test: digest: sha256:5d749d2b10150426b510d2c3a05a99cf547c2ca1be382e1dbb2f90b68b6bea96 size: 1362

docker run：启动一个容器

[root@DockerTestServer ~]# docker run -ti nginx bash

root@23bc7ccabb09:/#

在后台启动一个容器

[root@DockerTestServer ~]# docker run -tid nginx bash

1bcf5154d5c3a57d92a6796f526eac2cefd962aaca9cf4098689bfe830bb9e5e

映射本地1111端口到容器80端口

[root@DockerTestServer ~]# docker run -ti -p 1111:80 nginx bash

root@cd676d572188:/#

docker ps：查看当前运行的容器

docker ps -a：查看所有运行的容器

docker ps -q：正在运行的容器的ID

docker ps -aq：所有容器的ID

docker exec：进入到容器中

docker cp：拷贝文件到容器(双向拷贝)

docker rm：删除退出的容器

docker rmi：删除本机镜像

删掉所有停止的容器：docker rm `docker ps -aq`

docker tag：打版本号

docker commit：修改镜像信息并生成新镜像

### Dockerfile编写

FROM：继承基础镜像

MAINTAINER：镜像制作作者信息

RUN：用来执行shell命令

EXPOSE：暴露端口号

CMD：启动容器默认执行的命令

ENTRYPOINT：启动容器真正执行的命令

VOLUME：创建挂载点

ENV：配置环境变量

ADD：复制文件到容器

COPY：复制文件到容器

WORKDIR：设置容器的工作目录

USER：容器使用的用户

使用RUN命令创建用户

[root@DockerTestServer test]# cat Dockerfile

# base image

FROM centos:6

MAINTAINER dot

RUN useradd dot

执行构建

docker build -t centos:user .

ENV和CMD的使用

[root@DockerTestServer test]# cat Dockerfile

# base image

FROM centos:6

MAINTAINER dot

RUN useradd dot

RUN mkdir dot

ENV envir=test version=1.0

CMD echo "envir:$envir version:$version"

执行构建

docker build -t centos:env-cmd .

启动镜像验证env和cmd

[root@DockerTestServer test]# docker run centos:env-cmd

envir:test version:1.0

ADD一个压缩包会自动解压，workdir可以更改工作目录

# base image

FROM nginx

MAINTAINER dot

ADD ./index.tar.gz /usr/share/nginx/html/

WORKDIR /usr/share/nginx/html

COPY 拷贝指定目录下的所有文件到容器，不包括本目录

# base image

FROM nginx

MAINTAINER dot

ADD ./index.tar.gz /usr/share/nginx/html/

WORKDIR /usr/share/nginx/html

COPY webroot/ .

USER为启动容器的默认用户

# base image

FROM centos:6

MAINTAINER dot

ADD ./index.tar.gz /usr/share/nginx/html/

WORKDIR /usr/share/nginx/html

COPY webroot/ .

RUN useradd -m tomcat -u 1001

USER 1001

CMD和ENTRYPOINT的区别

如果在启动容器的时候自定义了执行的命令，会覆盖CMD

但是不会覆盖ENTRYPOINT，手动执行的命令会作为参数传递给ENTRYPOINT。

Volume 创建容器可挂载点

# base image

FROM centos:6

MAINTAINER dot

VOLUME /data

挂载web目录到/data

docker run -ti --rm -v `pwd`/web:/data centos:volume bash

宿主机目录必须写绝对路径

Alpine、scratch、Debian、slim