

dockerfile实践学习

需求：

让你安装一个mysql，且启动

如果是虚拟机部署形式的话，如下

1.开启vmware

2.运行某一个虚拟机，centos7

3.centos7安装mysql `yum install mysql-server`

4.通过脚本，或者命令，启动mysql即可

部署缓慢，且修改了宿主机环境，删除较为麻烦，占用宿主机一个3306端口

我们要学的是基于容器去运行mysql

1.开始vmware

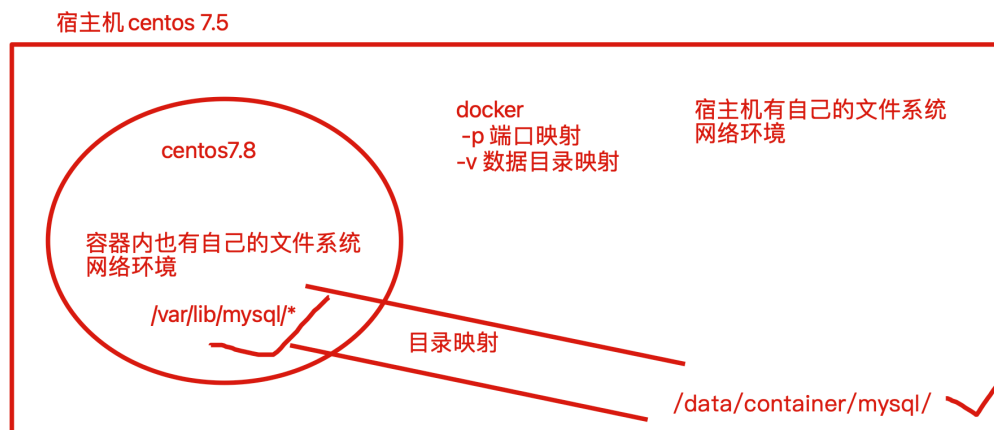
2.运行虚拟机centos7（宿主机）

3.安装docker容器软件

4.获取mysql镜像即可，`docker pull mysql:tag`（你无法自由控制，该mysql的基础镜像是什么发行版，你获取的镜像，是别人定制好，你下载使用的，debian，你希望得到一个基于centos7.8的发行版，运行mysql）

5.直接运行该镜像，通过端口映射，运行mysql，`docker run mysql:5.6`（容器能够运行，必须在容器内，有一个进程在前台运行，该容器内，有mysql正在前台运行）

6.访问宿主机的一个映射端口，访问到容器内的mysql



dockerfile实践

需求，通过dockerfile，构建nginx镜像，且运行容器后，生成的页面，是"超哥带你学习docker"

```
1 # 1. 创建Dockerfile, 注意文件名, 必须是这个
2 [root@yc_docker01 learn_docker]# pwd
3 /learn_docker
4 [root@yc_docker01 learn_docker]# cat Dockerfile
5 FROM nginx
6 RUN echo '<meta charset=utf8>超哥带你用docker运行nginx服务.' >
  /usr/share/nginx/html/index.html
7 [root@yc_docker01 learn_docker]#
8
9 # 2.构建Dockerfile
10 docker build .
11
12 # 3.修改镜像名字
13 [root@yc_docker01 learn_docker]# docker tag b4200a856253
  my_nginx
14
15 # 构建出的镜像如下
```

```
18
19 # 该容器运行时，执行的命令
20 # 等同于命令行的直接操作 docker run -it centos cat /etc/os-
  release
21 CMD ["cat", "/etc/os-release"]
22
23
24 # CMD有关启动程序的坑
25
26
```

把宿主机安装，启动nginx的理念放入到dockerfile

1. RUN yum install nginx
2. RUN 配置文件修改 sed
3. RUN systemctl start nginx ❌ 容器内的程序必须是前台运行，你的容器是启动不了的
4. 正确的应该是 CMD ["nginx", "-g", "daemon off;"]

docker面试题

ENTRYPOINT和CMD的区别以及用法

ENTRYPOINT和CMD的玩法

```
1 作用和CMD一样，都是在指定容器启动程序以及参数。
2
3 当指定了ENTRYPOINT之后，CMD指令的语义就有了变化
4 而是吧CMD的内容当作参数传递给ENTRYPOINT指令。
5
6 CMD    xxxx
7 ENTRYPOINT  xxx
8
```

```
41 就好比把该docker镜像，当做一个环境，去执行后面的命令
42 [root@yc_docker01 learn_docker]# docker run centos_curl pwd
43 /
44 [root@yc_docker01 learn_docker]#
45 [root@yc_docker01 learn_docker]# docker run centos_curl -I
46 docker: Error response from daemon: OCI runtime create
    failed: container_linux.go:380: starting container process
    caused: exec: "-I": executable file not found in $PATH:
    unknown.
47
48
49 # 7.想要正确的给容器传入一个 -I 参数该怎么办
50 希望容器内能正确完整，这个命令的执行
51 curl -s http://ipinfo.io/ip -I
52
53
54 # 8.解决办法1
55 给容器传入新的，完整的命令
56 # 这是投机取消的办法，不合适
57 [root@yc_docker01 learn_docker]# docker run centos_curl curl
    -s http://ipinfo.io/ip -I
58 HTTP/1.1 200 OK
59 access-control-allow-origin: *
60 content-type: text/html; charset=utf-8
61 content-length: 13
62 date: Tue, 29 Jun 2021 08:42:49 GMT
63 x-envoy-upstream-service-time: 1
64 Via: 1.1 google
65
66 # 9.正确的姿势应该是使用 ENTRYPOINT
67 修改dockerfile，如下
68
69 FROM centos:7.8.2003
70 RUN rpm --rebuilddb && yum install epel-release -y
```

```
101 date: Tue, 29 Jun 2021 08:46:23 GMT
102 x-envoy-upstream-service-time: 1
103 Via: 1.1 google
104
105
```

ARG和ENV指令

设置环境变量

容器1 nginx

容器2 php-fpm

容器3 mysql

```
1  dockerfile 脚本, shell脚本
2
3  ENV NAME="yuchao"
4  ENV AGE="18"
5  ENV MYSQL_VERSION=5.6
6
7  后续所有的操作, 通过$NAME 就可以直接获取变量值曹邹了, 维护
  dockerfile脚本时更友好, 方便
8  ADD
9  COPY
10 EXPOSE
11
12 ARG和ENV一样 设置环境变量
13 区别在于ENV 无论是在镜像构建时, 还是容器运行, 该变量都可以使用
14 ARG只是用于构建镜像需要设置的变量, 容器运行时就消失了
15
16
17
```