# 容器化编排部署mall-swarm

**1.1实战案例——容器化部署mall-swarm商城**

**1.1.1 案例目标**

（1）了解Mariadb服务的容器化部署。

（1）了解Redis服务的容器化部署。

（2）了解RabbitMQ服务的容器化部署。

（3）了解Nacos服务的容器化部署。

**1.1.2 案例分析**

**1.规划节点**

节点规划，见表1-1-1。

表1-1-1节点规划

|  |  |  |
| --- | --- | --- |
| **IP** | **主机名** | **节点** |
| 10.24.2.156 | master | docker-compose节点 |

**2.基础准备**

Docker和Docker Compose已安装完成，将提供的软件包mall-swarm.tar.gz上传至master节点/root目录下并解压。

**1.1.3 案例实施**

**1.Mariadb容器化部署**

（1）编写Dockerfile

编写yum文件

[root@master ~]# cd mall-swarm

[root@master mall-swarm]# cat local.repo

[mall]

name=mall

baseurl=file:///opt/mall-repo

gpgcheck=0

enabled=1

编写初始化脚本init.sh：

[root@master mall-swarm]# cat init.sh

#!/bin/bash

mysql\_install\_db --user=root

mysqld\_safe --user=root &

sleep 8

mysqladmin -u root password 'root'

mysql -uroot -proot -e "grant all on \*.\* to 'reader'@'%' identified by '123456'; flush privileges;"

mysql -uroot -proot -e "create database mall; use mall; source /opt/mall.sql;"

编写Dockerfile文件：

[root@master mall-swarm]# cat Dockerfile-mariadb

FROM centos:centos7.5.1804

MAINTAINER Guo

RUN rm -rf /etc/yum.repos.d/\*

COPY local.repo /etc/yum.repos.d/

COPY mall-repo /opt/mall-repo

COPY mall.sql /opt/

COPY init.sh /opt/

ENV LC\_ALL en\_US.UTF-8

RUN yum -y install mariadb-server && bash /opt/init.sh

EXPOSE 3306

CMD ["mysqld\_safe","--user=root"]

（2）构建镜像

构建镜像：

[root@master mall-swarm]# docker build -t mall-mysql:v1.0 -f Dockerfile-mariadb .

Sending build context to Docker daemon 410.3MB

Step 1/11 : FROM centos:centos7.5.1804

---> cf49811e3cdb

Step 2/11 : MAINTAINER Guo

---> Using cache

---> 612224da592d

Step 3/11 : RUN rm -rf /etc/yum.repos.d/\*

---> Using cache

---> d88b0763dfe4

Step 4/11 : COPY local.repo /etc/yum.repos.d/

---> Using cache

---> 60047baa075d

Step 5/11 : COPY mall-repo /opt/mall-repo

---> Using cache

---> ae7f67e9bf08

Step 6/11 : COPY mall.sql /opt/

---> Using cache

---> c387a39ef1aa

Step 7/11 : COPY init.sh /opt/

---> Using cache

---> a29b78a53fea

Step 8/11 : ENV LC\_ALL en\_US.UTF-8

---> Using cache

---> 19cf25317911

Step 9/11 : RUN yum -y install mariadb-server && bash /opt/init.sh

---> Using cache

---> 1e8fbec2cc00

Step 10/11 : EXPOSE 3306

---> Using cache

---> 9c4a6fb7bd6b

Step 11/11 : CMD ["mysqld\_safe","--user=root"]

---> Using cache

---> c689d4e7ea97

Successfully built c689d4e7ea97

Successfully tagged mall-mysql:v1.0

查看镜像列表：

[root@master mall-swarm]# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

mall-mysql v1.0 c689d4e7ea97 58 seconds ago 522MB

**2.Redis容器化部署**

（1）编写Dockerfile

编写Dockerfile文件：

[root@master mall-swarm]# cat Dockerfile-redis

FROM centos:centos7.5.1804

MAINTAINER Cuo

RUN rm -rf /etc/yum.repos.d/\*

COPY local.repo /etc/yum.repos.d/

COPY mall-repo /opt/mall-repo

RUN yum -y install redis

RUN sed -i 's/127.0.0.1/0.0.0.0/g' /etc/redis.conf && \

sed -i 's/protected-mode yes/protected-mode no/g' /etc/redis.conf

EXPOSE 6379

CMD ["/usr/bin/redis-server","/etc/redis.conf"]

（2）构建镜像

构建镜像：

[root@master mall-swarm]# docker build -t mall-redis:v1.0 -f Dockerfile-redis .

Sending build context to Docker daemon 410.3MB

Step 1/9 : FROM centos:centos7.5.1804

---> cf49811e3cdb

Step 2/9 : MAINTAINER Cuo

---> Using cache

---> dfdbec4d294b

Step 3/9 : RUN rm -rf /etc/yum.repos.d/\*

---> Using cache

---> cade39fa5cbd

Step 4/9 : COPY local.repo /etc/yum.repos.d/

---> Using cache

---> 83d77407ada3

Step 5/9 : COPY mall-repo /opt/mall-repo

---> Using cache

---> fa10113c62ac

Step 6/9 : RUN yum -y install redis

---> Using cache

---> 1b70d0d9faf5

Step 7/9 : RUN sed -i 's/127.0.0.1/0.0.0.0/g' /etc/redis.conf && sed -i 's/protected-mode yes/protected-mode no/g' /etc/redis.conf

---> Using cache

---> f9e69a29089d

Step 8/9 : EXPOSE 6379

---> Using cache

---> b45b59562c57

Step 9/9 : CMD ["/usr/bin/redis-server","/etc/redis.conf"]

---> Using cache

---> 44b9fefbf4e2

Successfully built 44b9fefbf4e2

Successfully tagged mall-redis:v1.0

查看镜像列表：

[root@master mall-swarm]# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

mall-redis v1.0 95746fc7acc8 54 seconds ago 338MB

**3.RabbitMQ容器化部署**

（1）编写Dockerfile

创建RabbitMQ用户脚本

[root@master mall-swarm]# cat rabbitmq-user.sh

#!/bin/bash

/usr/lib/rabbitmq/bin/rabbitmq-server restart

sleep 8

/usr/lib/rabbitmq/bin/rabbitmqctl add\_vhost mall

/usr/lib/rabbitmq/bin/rabbitmqctl add\_user mall mall

/usr/lib/rabbitmq/bin/rabbitmqctl set\_user\_tags mall administrator

/usr/lib/rabbitmq/bin/rabbitmqctl set\_permissions -p mall mall '.\*' '.\*' '.\*'

/usr/lib/rabbitmq/bin/rabbitmq-plugins enable rabbitmq\_management

/usr/lib/rabbitmq/bin/rabbitmq-server restart

编写Dockerfile文件：

[root@master mall-swarm]# cat Dockerfile-rabbitmq

FROM centos:centos7.5.1804

MAINTAINER Guo

RUN rm -rf /etc/yum.repos.d/\*

COPY local.repo /etc/yum.repos.d/

COPY rabbitmq-user.sh /opt/rabbitmq-user.sh

COPY mall-repo /opt/mall-repo

RUN yum -y install rabbitmq-server

EXPOSE 5672 15672

CMD ["/bin/bash","/opt/rabbitmq-user.sh"]

（2）构建镜像

构建镜像：

[root@master mall-swarm]# docker build -t mall-rabbit:v1.0 -f Dockerfile-rabbitmq .

Sending build context to Docker daemon 410.3MB

Step 1/9 : FROM centos:centos7.5.1804

---> cf49811e3cdb

Step 2/9 : MAINTAINER Guo

---> Using cache

---> 02855371e6e9

Step 3/9 : RUN rm -rf /etc/yum.repos.d/\*

---> Using cache

---> 5f56a1e30007

Step 4/9 : COPY local.repo /etc/yum.repos.d/

---> Using cache

---> ff9431065417

Step 5/9 : COPY rabbitmq-user.sh /opt/rabbitmq-user.sh

---> Using cache

---> d3d36333b8df

Step 6/9 : COPY mall-repo /opt/mall-repo

---> Using cache

---> 0efec0dea187

Step 7/9 : RUN yum -y install rabbitmq-server

---> Using cache

---> 918e0aea03a2

Step 8/9 : EXPOSE 5672 15672

---> Using cache

---> b796230efa8d

Step 9/9 : CMD ["/bin/bash","/opt/rabbitmq-user.sh"]

---> Using cache

---> 67d8b40d90ad

Successfully built 67d8b40d90ad

Successfully tagged mall-rabbit:v1.0

查看镜像列表：

[root@master mall-swarm]# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

mall-rabbit v1.0 67d8b40d90ad 3 minutes ago 374MB

**4.nacos-registry容器化部署**

（1）编写Dockerfile

编写启动服务脚本：

[root@master mall-swarm]# cat nacos-start.sh

#!/bin/bash

/usr/local/bin/nacos/bin/startup.sh -m standalone

tail -f /usr/local/bin/nacos/logs/start.out

编写Dockerfile文件：

# cat Dockerfile-nacos

FROM centos:centos7.5.1804

MAINTAINER Guo

COPY local.repo /etc/yum.repos.d/

COPY mall-repo /opt/mall-repo

COPY nacos-start.sh /opt

ADD jdk-8u121-linux-x64.tar.gz /usr/local/bin

ADD nacos-server-1.1.0.tar.gz /usr/local/bin

ENV JAVA\_HOME /usr/local/bin/jdk1.8.0\_121

EXPOSE 8848

CMD ["/bin/bash","/opt/ nacos-start.sh"]

（2）构建镜像

构建镜像：

[root@master mall-swarm]# docker build -t mall-nacos:v1.0 -f Dockerfile-nacos .

Sending build context to Docker daemon 410.3MB

Step 1/8 : FROM centos:centos7.5.1804

---> cf49811e3cdb

Step 2/8 : MAINTAINER Guo

---> Using cache

---> 02855371e6e9

Step 3/8 : COPY nacos-start.sh /opt

---> ee4c5c07528e

Step 4/8 : ADD jdk-8u121-linux-x64.tar.gz /usr/local/bin

---> ce589ce6c02c

Step 5/8 : ADD nacos-server-1.1.0.tar.gz /usr/local/bin

---> 35d35a2bd509

Step 6/8 : ENV JAVA\_HOME /usr/local/bin/jdk1.8.0\_121

---> Running in 3875e3c3e068

Removing intermediate container 3875e3c3e068

---> 879b73b899c8

Step 7/8 : EXPOSE 8848

---> Running in ed5c5e372f36

Removing intermediate container ed5c5e372f36

---> 57331b1d980e

Step 8/8 : CMD ["/bin/bash","/opt/nacos-start.sh"]

---> Running in 31b8f2fc713d

Removing intermediate container 31b8f2fc713d

---> ef7ddaa17cad

Successfully built ef7ddaa17cad

Successfully tagged mall-nacos:v1.0

查看镜像：

[root@master mall-swarm]# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

mall-nacos v1.0 ef7ddaa17cad 24 seconds ago 619MB

**5.容器化部署前端服务**

（1）编写Dockerfile

新建nginx文件夹：

生成前端文件：

[root@master mall-swarm]# tar -zxvf mall-admin-web.tar.gz

[root@master mall-swarm]# cd mall-admin-web

[root@master mall-admin-web]# vi config/prod.env.js

'use strict'

module.exports = {

NODE\_ENV: '"production"',

BASE\_API: '"http://10.24.2.156:8201/mall-admin"' #修改为本机IP

}

使用命令进行打包，生成dist目录：

[root@node mall-admin-web]# cd ../

[root@node mall-swarm]# tar zxvf node-v6.17.1-linux-x64.tar.gz

[root@node mall-swarm]# mv node-v6.17.1-linux-x64 /usr/local/node

[root@node mall-swarm]# vi /etc/profile

export NODE\_HOME=/usr/local/node

export PATH=$NODE\_HOME/bin:$PATH

[root@node mall-swarm]# source /etc/profile

[root@node mall-swarm]# node -v

v6.17.1

[root@node mall-swarm]# npm -v

3.10.10

[root@node mall-swarm]# cd mall-admin-web

[root@master mall-admin-web]# npm run build

[root@master mall-admin-web]# mv dist/ ../

[root@master mall-admin-web]# cd ../

编写Dockerfile：

[root@master mall-swarm]# cat Dockerfile-nginx

FROM centos:centos7.5.1804

MAINTAINER Guo

RUN rm -rf /etc/yum.repos.d/\*

COPY local.repo /etc/yum.repos.d/

COPY mall-repo /opt/mall-repo

RUN yum -y install nginx

COPY dist/ /usr/share/nginx/html/

EXPOSE 80

CMD ["nginx","-g","daemon off;"]

（2）构建镜像

构建镜像：

[root@master mall-swarm]# docker build -t mall-nginx:v1.0 -f Dockerfile-nginx .

Sending build context to Docker daemon 700.1MB

Step 1/9 : FROM centos:centos7.5.1804

---> cf49811e3cdb

Step 2/9 : MAINTAINER Guo

---> Using cache

---> 02855371e6e9

Step 3/9 : RUN rm -rf /etc/yum.repos.d/\*

---> Using cache

---> 5f56a1e30007

Step 4/9 : COPY local.repo /etc/yum.repos.d/

---> Using cache

---> ff9431065417

Step 5/9 : COPY mall-repo /opt/mall-repo

---> Using cache

---> 0a63c0b0b053

Step 6/9 : RUN yum -y install nginx

---> Using cache

---> 69686a3a123e

Step 7/9 : COPY dist/ /usr/share/nginx/html/

---> Using cache

---> 1b12b034cef7

Step 8/9 : EXPOSE 80

---> Using cache

---> 3435ab9920cb

Step 9/9 : CMD ["nginx","-g","daemon off;"]

---> Using cache

---> 758615ecb15e

Successfully built 758615ecb15e

Successfully tagged mall-nginx:v1.0

查看镜像列表：

root@master mall-swarm]# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

mall-nginx v1.0 758615ecb15e About a minute ago 434MB

**6.编排部署Mall-Swarm**

（1）编写docker-compose.yaml

导入镜像：

[root@master mall-swarm]# docker load -i images/mall\_mall-admin\_1.0-SNAPSHOT.tar

[root@master mall-swarm]# docker load -i images/mall\_mall-auth\_1.0-SNAPSHOT.tar

[root@master mall-swarm]# docker load -i images/mall\_mall-gateway\_1.0-SNAPSHOT.tar

编写docker-compose.yaml：

[root@master mall-swarm]# cat docker-compose.yaml

version: '3'

services:

mysql:

image: mall-mysql:v1.0

container\_name: mysql

restart: always

ports:

- 3306:3306

redis:

image: mall-redis:v1.0

container\_name: redis

ports:

- 6379:6379

nginx:

image: mall-nginx:v1.0

container\_name: nginx

ports:

- 8888:80

rabbitmq:

image: mall-rabbit:v1.0

container\_name: rabbitmq

ports:

- 5672:5672

- 15672:15672

nacos-registry:

image: mall-nacos:v1.0

container\_name: nacos-registry

ports:

- 8848:8848

mall-admin:

image: mall/mall-admin:1.0-SNAPSHOT

container\_name: mall-admin

ports:

- 8080:8080

links:

- mysql:db

mall-gateway:

image: mall/mall-gateway:1.0-SNAPSHOT

container\_name: mall-gateway

ports:

- 8201:8201

links:

- redis:redis

- nacos-registry:nacos-registry

mall-auth:

image: mall/mall-auth:1.0-SNAPSHOT

container\_name: mall-auth

ports:

- 8401:8401

links:

- nacos-registry:nacos-registry

（2）部署服务

启动服务：

[root@master mall-swarm]# cd /opt/harbor/harbor/

[root@master mall-swarm]# docker-compose down

[root@master mall-swarm]# cd -

[root@master mall-swarm]# docker-compose up -d

Creating redis ... done

Creating rabbitmq ... done

Creating mysql ... done

Creating nginx ... done

Creating nacos-registry ... done

Creating mall-gateway ... done

Creating mall-auth ... done

Creating mall-admin ... done

[root@master mall-swarm]# docker-compose ps

Name Command State Ports

---------------------------------------------------------------------------------------------------------

mall-admin java -jar -Dspring.profile ... Up 0.0.0.0:8080->8080/tcp

mall-auth java -jar -Dspring.profile ... Up 0.0.0.0:8401->8401/tcp

mall-gateway java -jar -Dspring.profile ... Up 0.0.0.0:8201->8201/tcp

mysql mysqld\_safe --user=root Up 0.0.0.0:3306->3306/tcp

nacos-registry /bin/bash /opt/nacos-start.sh Up 0.0.0.0:8848->8848/tcp

nginx nginx -g daemon off; Up 0.0.0.0:8888->80/tcp

rabbitmq /bin/bash /opt/rabbitmq-us ... UP 0.0.0.0:15672->15672/tcp, 0.0.0.0:5672->5672/tcp

redis /usr/bin/redis-server /etc ... Up 0.0.0.0:6379->6379/tcp

查看：

# docker-compose ps

Name Command State Ports

----------------------------------------------------------------------------------------------------------

mall-admin java -jar -Dspring.profile ... Up 0.0.0.0:8080->8080/tcp

mall-auth java -jar -Dspring.profile ... Up 0.0.0.0:8401->8401/tcp

mall-gateway java -jar -Dspring.profile ... Up 0.0.0.0:8201->8201/tcp

mysql mysqld\_safe --user=root Up 0.0.0.0:3306->3306/tcp

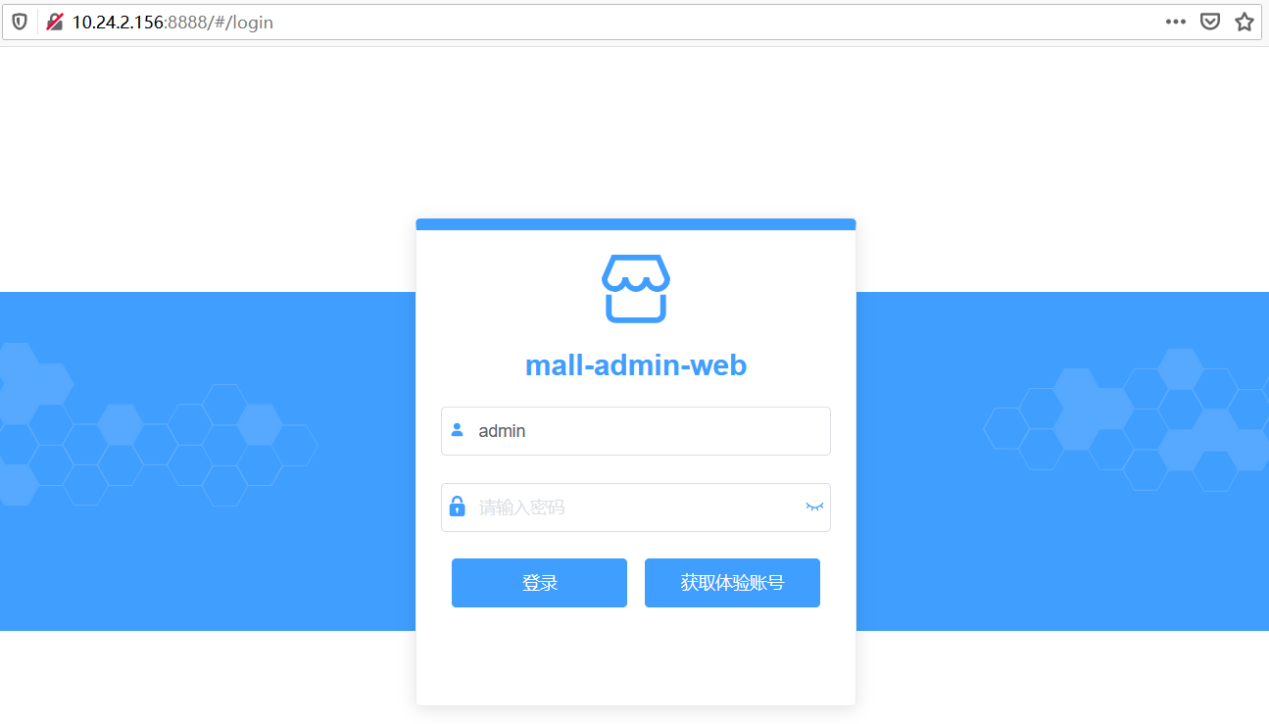
nacos-registry /bin/bash /opt/run.sh Up 0.0.0.0:8848->8848/tcp

nginx nginx -g daemon off; Up 0.0.0.0:8888->80/tcp

rabbitmq /bin/bash /opt/run.sh Up 0.0.0.0:15672->15672/tcp, 0.0.0.0:5672->5672/tcp

redis /usr/bin/redis-server /etc ... Up 0.0.0.0:6379->6379/tcp

在浏览器上通过http://IP:8888访问mall-swarm，如图所示：



输入登录信息（admin/macro123）如图所示：

