



链滴

# 三 Linux 主机 配置 docker 一键启动 rocke tMQ 集群

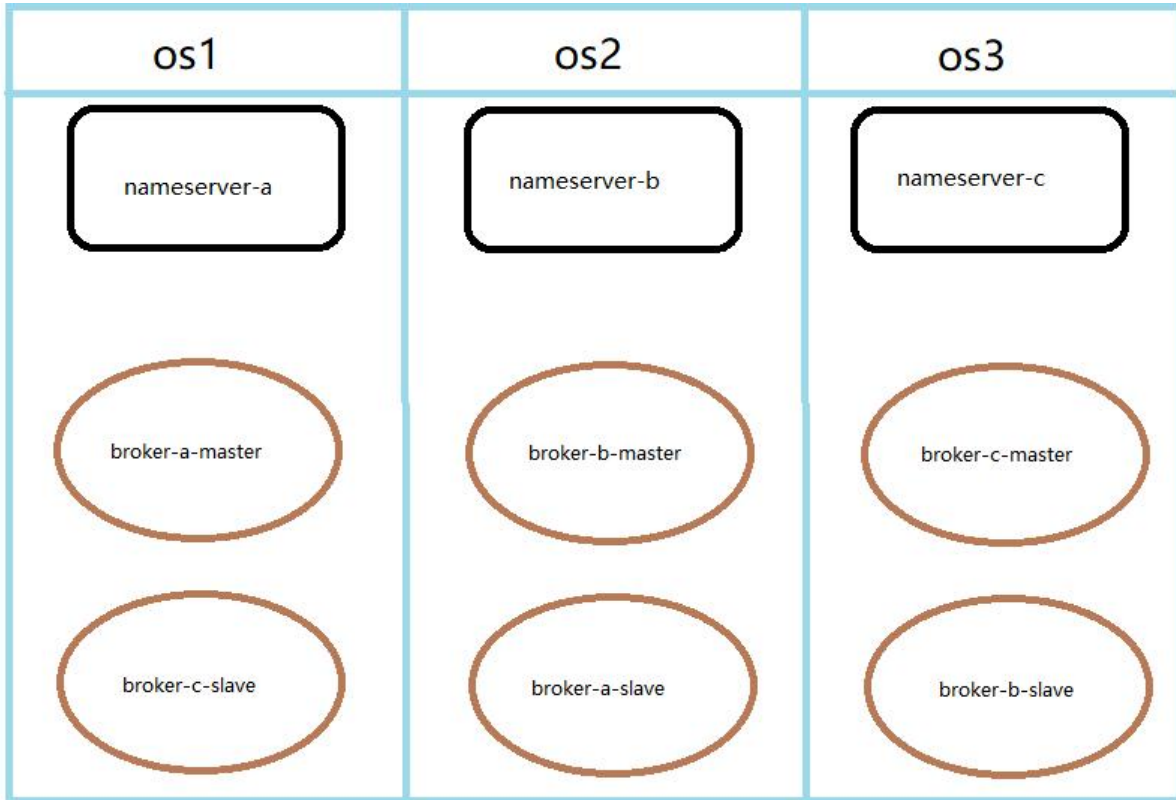
作者: [Hawkpool](#)

原文链接: <https://ld246.com/article/1576129125031>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

本次使用docker配置三master三slave的rocketmq集群



## 开工前准备:

三台centos 服务器/虚拟机

[搭建docker+docker-compose环境](#)

开放防火墙(或者自行开放相应端口)

## 配置rocket本地映射目录

运行脚本

三台机器分别运行脚本

```
mkdir -p /opt/rocketmq/logs/nameserver-1
mkdir -p /opt/rocketmq/logs/nameserver-2
mkdir -p /opt/rocketmq/store/nameserver-1
mkdir -p /opt/rocketmq/store/nameserver-2
mkdir -p /opt/rocketmq/logs/broker-1
mkdir -p /opt/rocketmq/logs/broker-2
mkdir -p /opt/rocketmq/store/broker-1
mkdir -p /opt/rocketmq/store/broker-2
mkdir -p /home/rocketmq/broker-1
mkdir -p /home/rocketmq/broker-2
```

```
mkdir -p /opt/rocketmq/logs/nameserver-3
mkdir -p /opt/rocketmq/logs/nameserver-4
mkdir -p /opt/rocketmq/store/nameserver-3
```

```
mkdir -p /opt/rocketmq/store/nameserver-4
mkdir -p /opt/rocketmq/logs/broker-3
mkdir -p /opt/rocketmq/logs/broker-4
mkdir -p /opt/rocketmq/store/broker-3
mkdir -p /opt/rocketmq/store/broker-4
mkdir -p /home/rocketmq/broker-3
mkdir -p /home/rocketmq/broker-4
```

```
mkdir -p /opt/rocketmq/logs/nameserver-5
mkdir -p /opt/rocketmq/logs/nameserver-6
mkdir -p /opt/rocketmq/store/nameserver-5
mkdir -p /opt/rocketmq/store/nameserver-6
mkdir -p /opt/rocketmq/logs/broker-5
mkdir -p /opt/rocketmq/logs/broker-6
mkdir -p /opt/rocketmq/store/broker-5
mkdir -p /opt/rocketmq/store/broker-6
mkdir -p /home/rocketmq/broker-5
mkdir -p /home/rocketmq/broker-6
```

## 分别配置三台机器的 broker.conf

配置文件的文件夹存放在 /home/rocketmq 目录下

以os1的配置为例

master

```
brokerClusterName = rocketmq-cluster
brokerName = broker-a-master
brokerId = 1
#这个很有讲究 如果是正式环境 这里一定要填写内网地址 (安全)
#如果是用于测试或者本地这里建议要填外网地址, 因为你的本地代码是无法连接到阿里云内网, 只连接外网。
brokerIP1 = 10.10.20.197
deleteWhen = 04
fileReservedTime = 48
brokerRole = ASYNC_MASTER
flushDiskType = ASYNC_FLUSH
# 内网的(阿里云有内网IP和外网IP)
namesrvAddr=10.10.20.197:9876;10.10.20.198:9876;10.10.20.199:9876
autoCreateTopicEnable=true
#Broker 对外服务的监听端口,
listenPort = 10911
#Broker角色
#- ASYNC_MASTER 异步复制Master
#- SYNC_MASTER 同步双写Master
#- SLAVE
brokerRole=ASYNC_MASTER
#刷盘方式
#- ASYNC_FLUSH 异步刷盘
#- SYNC_FLUSH 同步刷盘
flushDiskType=ASYNC_FLUSH
```

slave

```
brokerClusterName = rocketmq-cluster
brokerName = broker-c-slave
brokerId = 2
brokerIP1 = 10.10.20.197
deleteWhen = 04
fileReservedTime = 48
brokerRole = ASYNC_MASTER
flushDiskType = ASYNC_FLUSH
# 内网的(阿里云有内网IP和外网IP)
namesrvAddr=10.10.20.197:9876;10.10.20.198:9876;10.10.20.199:9876
autoCreateTopicEnable=true
#Broker 对外服务的监听端口,
listenPort = 10909
#Broker角色
#- ASYNC_MASTER 异步复制Master
#- SYNC_MASTER 同步双写Master
#- SLAVE
brokerRole=SLAVE
#刷盘方式
#- ASYNC_FLUSH 异步刷盘
#- SYNC_FLUSH 同步刷盘
flushDiskType=ASYNC_FLUSH
```

## 编写 docker-compose.yml

```
cd /home/rocketmq
touch docker-compose.yml
vi docke-compose.yml
```

```
version: '3.5'
services:
  rmqnamesrv-a:
    image: rocketmqinc/rocketmq:4.3.0
    container_name: rmqnamesrv-a
    ports:
      - 9876:9876
    volumes:
      - /opt/rocketmq/logs/nameserver-a:/opt/logs
      - /opt/rocketmq/store/nameserver-a:/opt/store
    command: sh mqnamesrv
    networks:
      rmq:
    aliases:
      - rmqnamesrv-a

  rmqbroker-a-master:
    image: rocketmqinc/rocketmq:4.3.0
    container_name: rmqbroker-a-master
    ports:
      - 10911:10911
    volumes:
      - /opt/rocketmq/logs/broker-1:/opt/logs
      - /opt/rocketmq/store/broker-1:/opt/store
```

```
- /home/rocketmq/broker-1/broker-1.conf:/opt/rocketmq-4.3.0/conf/broker.conf
environment:
TZ: Asia/Shanghai
NAMESRV_ADDR: "10.10.20.197:9876;10.10.20.198:9876;10.10.20.199:9876"
JAVA_OPTS: "-Duser.home=/opt"
JAVA_OPT_EXT: "-server -Xms256m -Xmx256m -Xmn256m"
command: sh mqbroker -c /opt/rocketmq-4.3.0/conf/broker.conf autoCreateTopicEnable=true &
links:
- rmqnamesrv-a:rmqnamesrv-a
networks:
rmq:
aliases:
- rmqbroker-a-master
```

```
rmqbroker-c-slave:
image: rocketmqinc/rocketmq:4.3.0
container_name: rmqbroker-c-slave
ports:
- 10909:10909
volumes:
- /opt/rocketmq/logs/broker-2:/opt/logs
- /opt/rocketmq/store/broker-2:/opt/store
- /home/rocketmq/broker-2/broker-2.conf:/opt/rocketmq-4.3.0/conf/broker.conf
environment:
TZ: Asia/Shanghai
NAMESRV_ADDR: "10.10.20.197:9876;10.10.20.198:9876;10.10.20.199:9876"
JAVA_OPTS: "-Duser.home=/opt"
JAVA_OPT_EXT: "-server -Xms256m -Xmx256m -Xmn256m"
command: sh mqbroker -c /opt/rocketmq-4.3.0/conf/broker.conf autoCreateTopicEnable=true &
links:
- rmqnamesrv-a:rmqnamesrv-a
networks:
rmq:
aliases:
- rmqbroker-c-slave
rmqconsole:
image: styletang/rocketmq-console-ng
container_name: rmqconsole
ports:
- 8080:8080
environment:
JAVA_OPTS: -Drocketmq.namesrv.addr=10.10.20.197:9876;10.10.20.198:9876;10.10.20.199:9876 -Dcom.rocketmq.sendMessageWithVIPChannel=false
networks:
rmq:
aliases:
- rmqconsole
networks:
rmq:
name: rmq
driver: bridge
```

通过Docker总共拉取了4条镜像记录。**rmqnamesrv-a**、**rmqbroker-a-master**、**rmqbroker-c-slave**、**rmqconsole**

**rmqconsole**是一个可视化的工具，可以通过页面来查看RocketMQ相关信息

可以看出对于broker的配置文件broker.conf已经被broker-1/2.conf替换

全部配置文件可以参考下载文件 [📄heart](#)

[下载链接](#) 提取码: **d54d**

## 一键启动

进入 /home/rocketmq 目录

```
docker-compose -f docker-compose.yml up -d
```

启动成功可以进入 <http://OS1/2/3的ip:8080/#/>

查看是否成功集群



The screenshot shows the RocketMQ Control Console interface. At the top, there is a navigation bar with tabs for '集群' (Cluster), '主题' (Topic), '消费者' (Consumer), and '生产者' (Producer). Below the navigation bar, there is a dropdown menu for the cluster name, currently set to 'rocketmq-cluster'. The main content area displays a table with the following columns: '分片' (Shard), '编号' (ID), '地址' (Address), '版本' (Version), '生产消息TPS' (Production Message TPS), '消费消息TPS' (Consumption Message TPS), '昨日生产总数' (Yesterday's Total Production), '昨日消费总数' (Yesterday's Total Consumption), '今天生产总数' (Today's Total Production), '今天消费总数' (Today's Total Consumption), and '操作' (Operations). The table lists six nodes: broker-b-master (ID 0), broker-b-slave (ID 6), broker-c-master (ID 0), broker-a-master (ID 0), broker-a-slave (ID 4), and broker-c-slave (ID 2). All nodes are running version V4\_3\_0 and show 0.00 TPS for both production and consumption. The '操作' column contains '状态' (Status) and '配置' (Configuration) buttons for each node.

分片	编号	地址	版本	生产消息TPS	消费消息TPS	昨日生产总数	昨日消费总数	今天生产总数	今天消费总数	操作
broker-b-master	0(master)	10.10.20.198.10911	V4_3_0	0.00	0.00	0	0	0	0	状态 配置
broker-b-slave	6(slave)	10.10.20.199.10909	V4_3_0	0.00	0.00	0	0	0	0	状态 配置
broker-c-master	0(master)	10.10.20.199.10911	V4_3_0	0.00	0.00	0	0	0	0	状态 配置
broker-a-master	0(master)	10.10.20.197.10911	V4_3_0	0.00	0.00	0	0	0	0	状态 配置
broker-a-slave	4(slave)	10.10.20.198.10909	V4_3_0	0.00	0.00	0	0	0	0	状态 配置
broker-c-slave	2(slave)	10.10.20.197.10909	V4_3_0	0.00	0.00	0	0	0	0	状态 配置