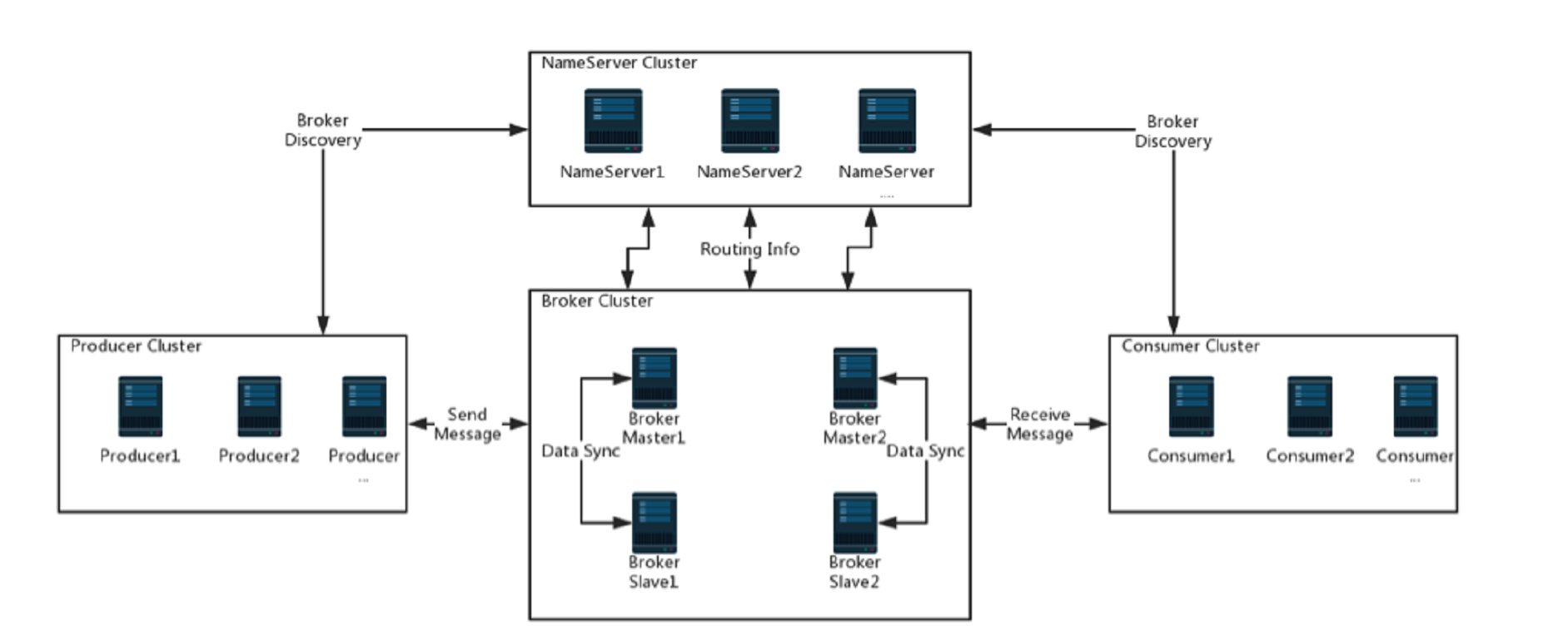
消费组与生产组需要统一，否则若是有两个消费组，当一个消费组消费掉消息之后，另一个消费组启动重新开始消费消息

Rocketmq的broker slave 不能写入数据，消息只能发送到master

Rcoketmq延迟队列的实现，使用中间队列的方式，消息先发送到在中间队列，然后由单线程能对队列进行扫描，将到期的信息发送到业务队列，由消费者进行消费，此处存在一个问题，如果生产者生产消息速度很快，则可能造成数据延期问题

# 1、rocketMQ的安装



Apache RocketMQ是一个分布式消息传递和流媒体平台， 具有低延迟， 高性能和可靠性， 万亿级容量和灵活的可伸缩性。 它由四个部分组成： nameserver， broker， 生产者和使用 者。 它们中的每一个都可以水平扩展， 而没有单个故障点。 如上面的屏幕截图所示。 nameserver： 提供轻量级的服务发现和路由。 每个名称服务器记录完整的路由信息， 提供 相应的读写服务， 并支持快速的存储扩展。 Broker： 通过提供轻量级的TOPIC和QUEUE机制来存储消息

下载地址： http://mirrors.tuna.tsinghua.edu.cn/apache/rocketmq/4.3.2/rocketmq­all-  
4.3.2­bin­release.zip

## 1.1 rocketmq单机版安装

1、加入rocketmq环境变量

export ROCKETMQ\_HOME=/usr/local/rocketmq/rocketmq‐all‐4. 1. 0‐incubating  
export PATH=$ROCKETMQ\_HOME/bin: $PATH

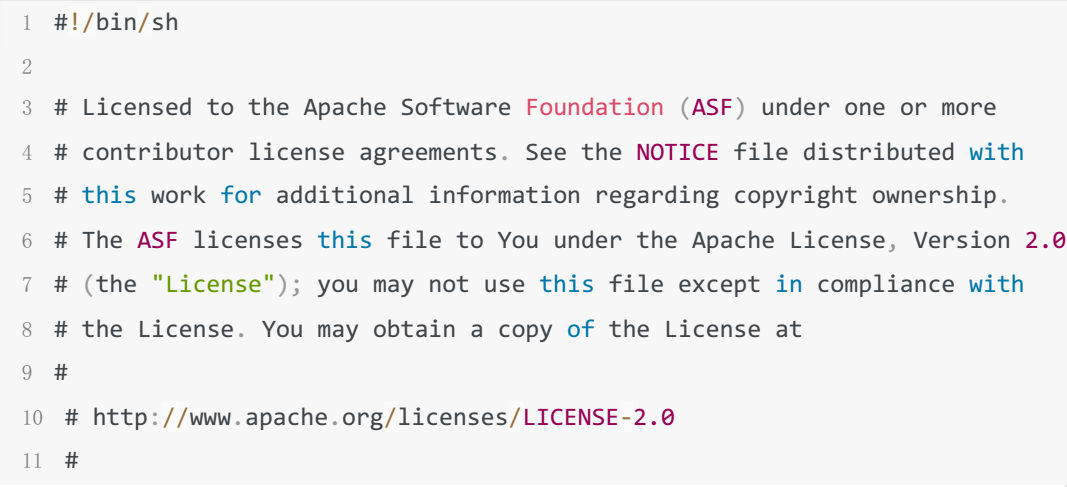
2、刷新环境变量

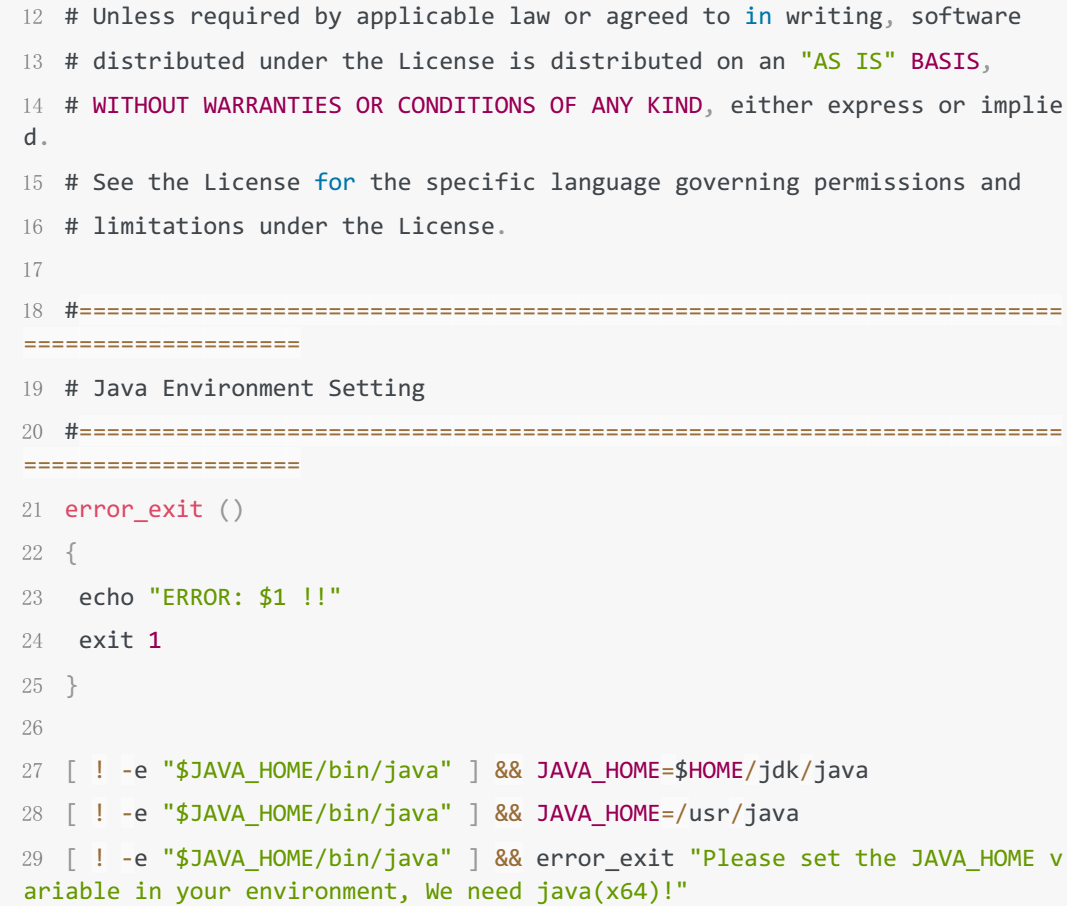
刷新环境变量： source /etc/profile

3、修改/usr/local/rocketmq/conf/broker.conf(启动时可选择该配置文件)

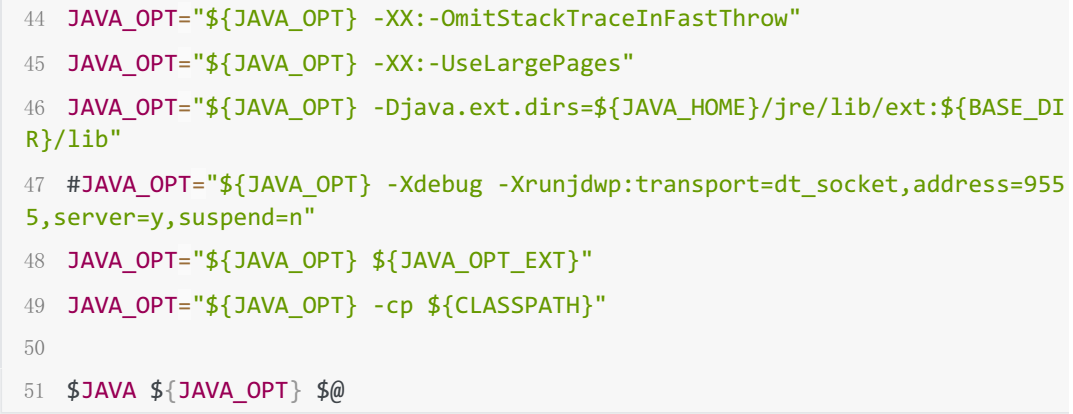


4、机器内存不够(一般针对虚拟机)， 修改/usr/local/rocketmq/bin/runserver.sh 与  
/usr/local/rocketmq/bin/runbroker.sh中JAVA\_OPT关于内存的设置  
执行命令： vim runserver.sh

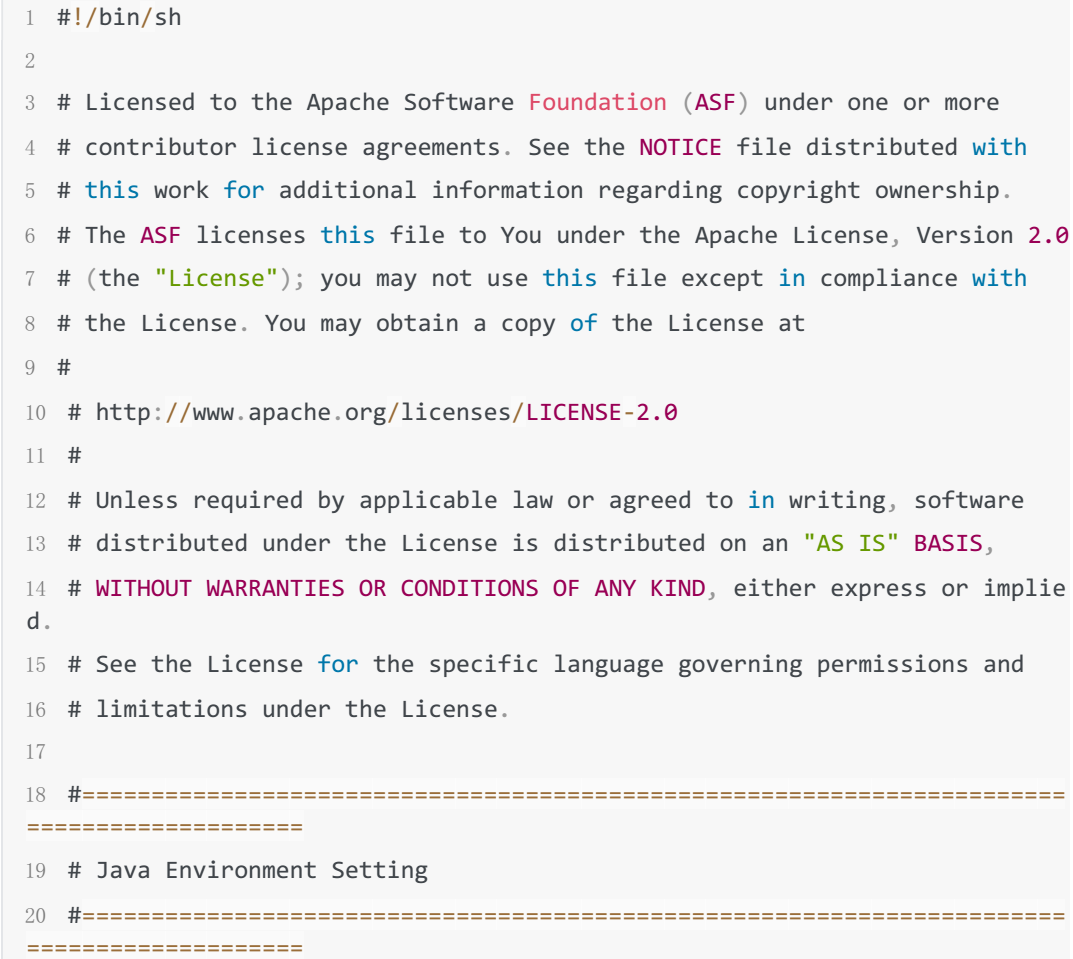


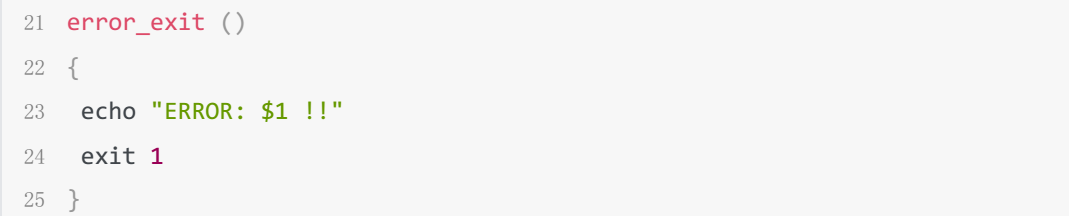






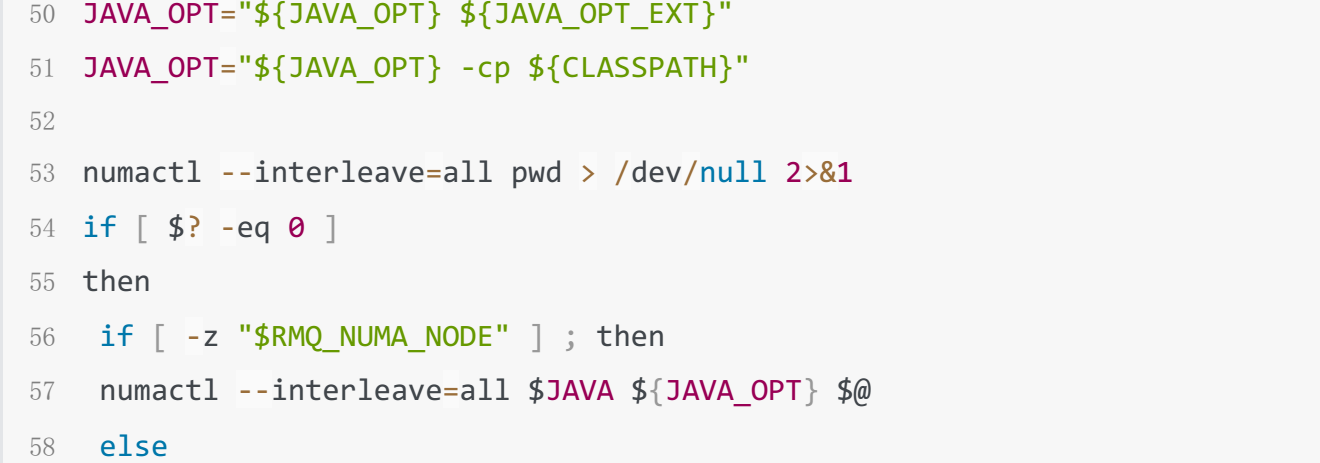
执行命令： vim runbroker.sh

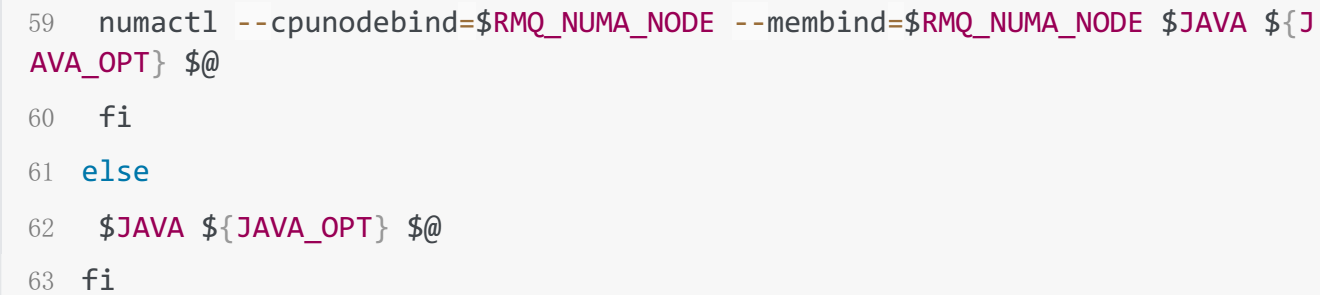












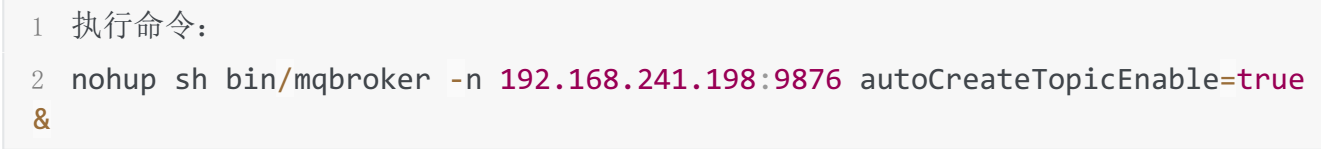
5、修改配置文件broker.conf的ip地址



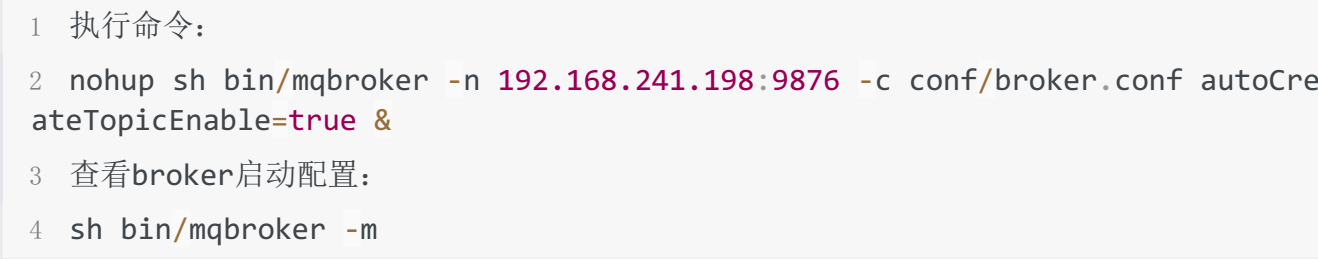
6、运行服务namesrv（需在$ROCKETMQ\_HOME/bin下执行）



7、启动broker（需在$ROCKETMQ\_HOME/bin下执行）  
不指定启动配置文件

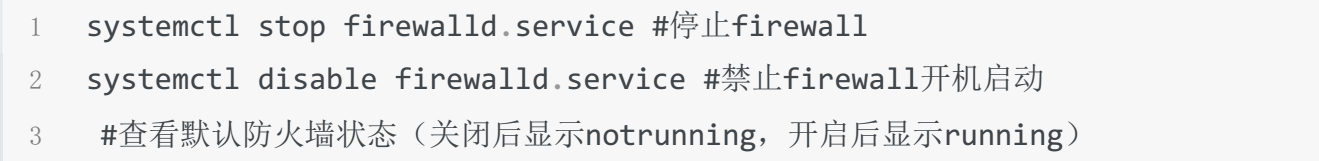


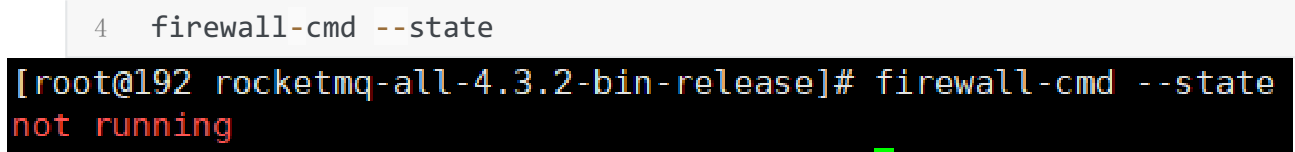
指定配置文件启动



（1 27. 0. 0. 1 : 9876为nameserver， 链接进行注册， autoCreateTopicEnable=true(允许创建topic);)

8、客户端访问可能会出现的问题：  
RemotingTooMuchRequestException: sendDefaultImpl call timeout  
在客户端运行Producer时， 可能会出现如上异常， 这是因为从 Windows 上开发连接  
虚拟机中的 nameServer 时要经过 Linux 系统的防火墙， 而防火墙一般都会有超时的机  
制， 在网络连接长时间不传输数据时， 会关闭这个 TCP 的会话， 关闭后再读写， 就有可能导致这个异常。  
contOS下命令如下：





9、关闭命令

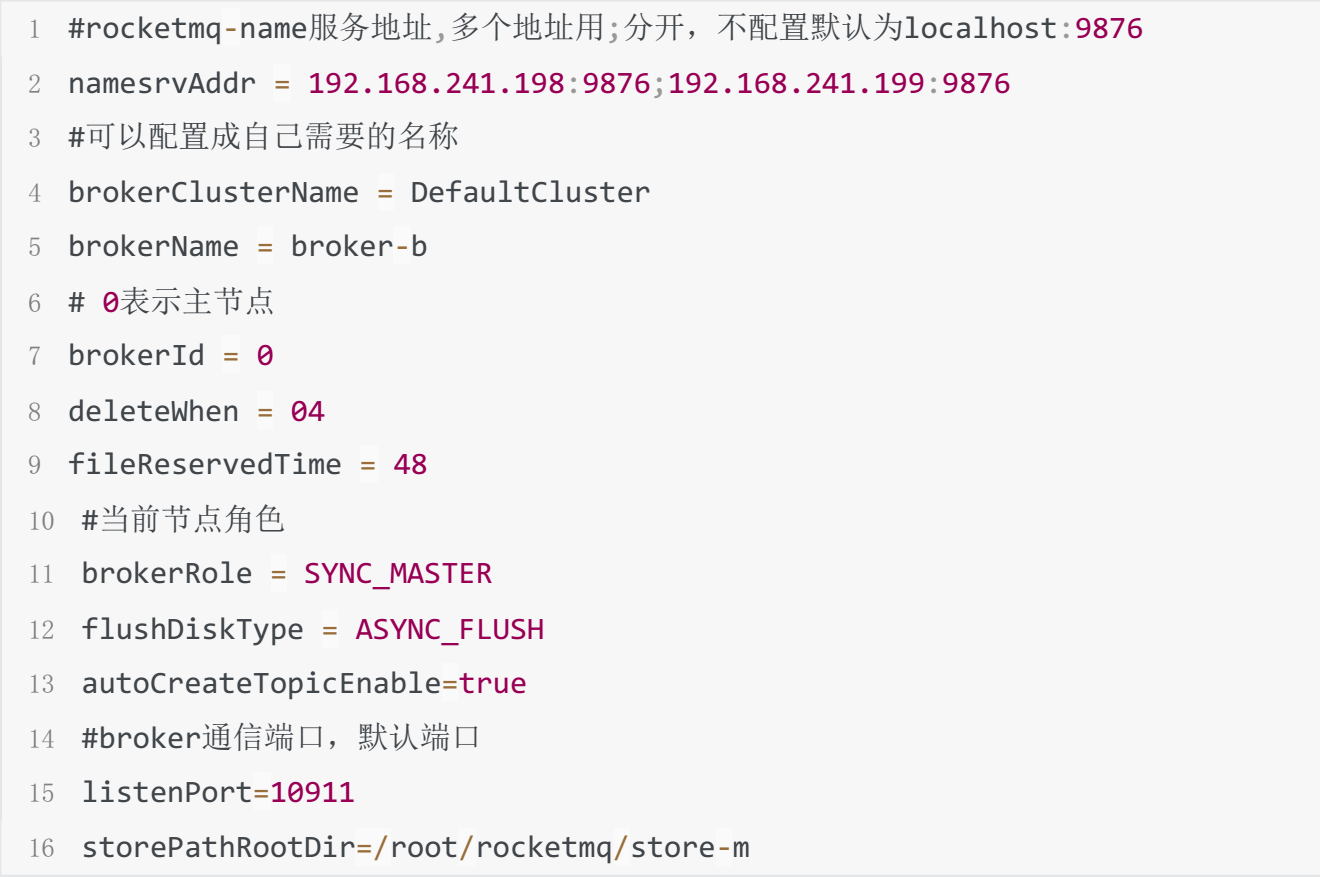


## 1.2 rocketmq集群部署

为保证高可用，rocketmq部署时使用互为主备，首先只有master broker可以接受客户端消息，然后同步给slave broker，互为主备即为在一个节点下部署一个master-01节点，然后在启动一个slave-02的master-02的从节点，这样在另一台机器上的master-02所在服务器挂掉之后，其他服务器上还会存在一个slave broker，虽然其不能转变为master，但是其可以保证读消息功能仍然是正常的

准备2个虚拟机分别是虚拟机centos-node-01 与centos-node-02， 分别部署2个NameServer， 并在每台机器上分别启动一个Master和一个Slave， 互为主备， 在主目 录  
下的conf文件夹下提供了多种broker配置模式， 分别有： 2m-2s-async， 2m-2ssync， 2m-noslave， 可以以此为模版做如下配置：  
cd conf目 录  
cp broker.conf broker-m.conf  
cp broker.conf broker-s.conf

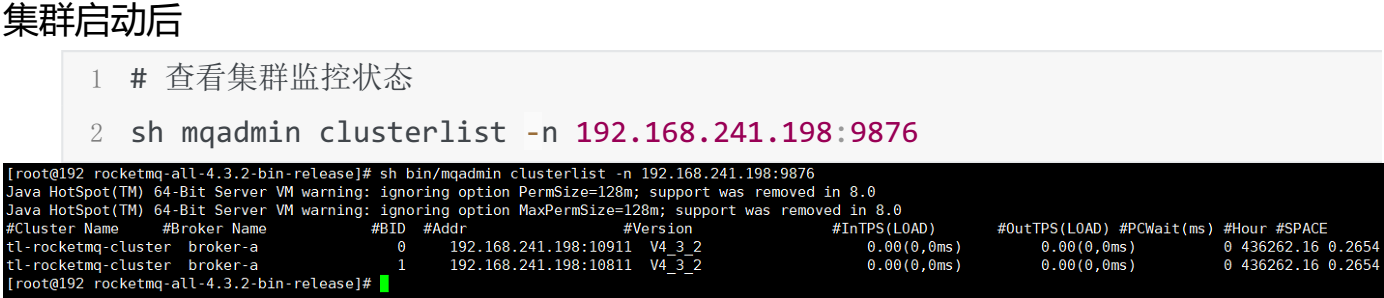
Master conf配置

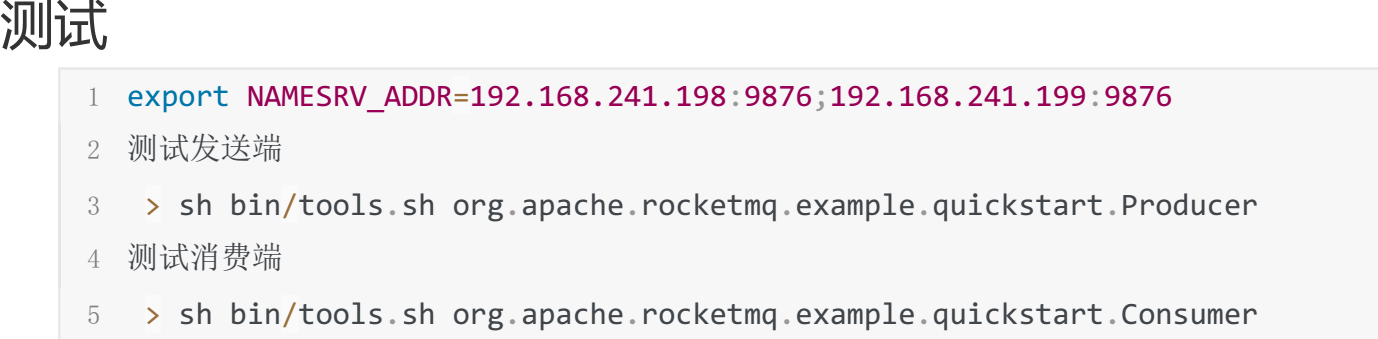


Slave conf配置









配置说明

