1. 将zookeeper集群启动起来
2. 修改kafka根目录下的config/server.properties

broker.id=1 #整个集群内唯一id号，整数，一般从0开始

listeners=PLAINTEXT://192.168.49.10:9092 #协议、当前broker机器ip、端口，此值可以配置多个，应该跟SSL等有关系，更多用法尚未弄懂，这里修改为ip和端口。

log.dirs=/tmp/kafka/log1 #kafka存储日志的目录

zookeeper.connect=192.168.49.10:2181,192.168.49.10:2182,192.168.49.10:2183 #zookeeper 集群列表

broker.id=2 #整个集群内唯一id号，整数，一般从0开始

listeners=PLAINTEXT://192.168.49.10:9093 #协议、当前broker机器ip、端口，此值可以配置多个，应该跟SSL等有关系，更多用法尚未弄懂，这里修改为ip和端口。

log.dirs=/tmp/kafka/log2 #kafka存储日志的目录

zookeeper.connect=192.168.49.10:2181,192.168.49.10:2182,192.168.49.10:2183 #zookeeper 集群列表

broker.id=3 #整个集群内唯一id号，整数，一般从0开始

listeners=PLAINTEXT://192.168.49.10:9094 #协议、当前broker机器ip、端口，此值可以配置多个，应该跟SSL等有关系，更多用法尚未弄懂，这里修改为ip和端口。

log.dirs=/tmp/kafka/log3 #kafka存储日志的目录

zookeeper.connect=192.168.49.10:2181,192.168.49.10:2182,192.168.49.10:2183 #zookeeper 集群列表

1. kafka集群启动

./kafka-server-start.sh -daemon ../config/server1.properties

./kafka-server-start.sh -daemon ../config/server2.properties

./kafka-server-start.sh -daemon ../config/server3.properties

1. 创建主题

./kafka-topics.sh --create --zookeeper 192.168.49.10:2181,192.168.49.10:2182,192.168.49.10:2183 --replication-factor 3 --partitions 3 --topic test

1. 查看test详情

./kafka-topics.sh --describe --zookeeper 192.168.49.10:2181 --topic test

1. 启动生产者

./kafka-console-producer.sh --broker-list 192.168.49.10:9092 --topic test

1. 启动消费者

./kafka-console-consumer.sh --bootstrap-server 192.168.49.10:9092 --topic test --from-beginning

8、模拟kafka broker集群某个节点宕机，直接通过kill 进程来模拟，观察生产者和消费者某个broker节点宕机后的生产者消费者，我们可以发现，消费者出现了警告信息，但是依旧可以消费数据。

9、模拟zookeeper集群某个节点宕机，直接通过kill 进程来模拟，观察生产者和消费者某个zookeeper节点宕机后的生产者消费者，我们可以发现，消费者出现了警告信息，但是依旧可以消费数据。