# 简要说明

本教程使用CentOS release 6.5 (Final)、kafka\_2.10-0.10.0.0、jdk 1.8.0\_172（本文省略了jdk的安装教程）。

本教程一共三个zookeeper实例，其中两个搭建在172.16.15.6机器上，另外一个搭建在172.16.15.7机器上。

# 源码下载

从官网下载

https://archive.apache.org/dist/kafka/0.10.0.0/kafka\_2.10-0.10.0.0.tgz

将kafka\_2.10-0.10.0.0.tgz拷贝到/usr/local下

# 解压源码

tar -zxvf kafka\_2.10-0.10.0.0.tgz

# 进入到kafka的config目录

cd /usr/local/kafka\_2.10-0.10.0.0/config/

# 配置zookeeper启动参数

[root@dianliangcaiji139 ~]# cd /usr/local/kafka\_2.10-0.10.0.0/config/

[root@dianliangcaiji139 config]# pwd

/usr/local/kafka\_2.10-0.10.0.0/config

[root@dianliangcaiji139 config]# vi zookeeper.properties



#server.1 这个1是服务器的标识也可以是其他的数字， 表示这个是第几号服务器，用来标识服务器，这个标识要写到快照目录下面myid文件里

#192.168.7.107为集群里的IP地址，第一个端口是master和slave之间的通信端口，默认是2888，第二个端口是leader选举的端口，集群刚启动的时候选举或者leader挂掉之后进行新的选举的端口默认是3888

#tickTime：

这个时间是作为 Zookeeper 服务器之间或客户端与服务器之间维持心跳的时间间隔，也就是每个 tickTime 时间就会发送一个心跳。

#initLimit：

这个配置项是用来配置 Zookeeper 接受客户端（这里所说的客户端不是用户连接 Zookeeper 服务器的客户端，而是 Zookeeper 服务器集群中连接到 Leader 的 Follower 服务器）初始化连接时最长能忍受多少个心跳时间间隔数。当已经超过 5个心跳的时间（也就是 tickTime）长度后 Zookeeper 服务器还没有收到客户端的返回信息，那么表明这个客户端连接失败。总的时间长度就是 5\*2000=10 秒

#syncLimit：

这个配置项标识 Leader 与Follower 之间发送消息，请求和应答时间长度，最长不能超过多少个 tickTime 的时间长度，总的时间长度就是5\*2000=10秒

#dataDir：

快照日志的存储路径

#dataLogDir：

事物日志的存储路径，如果不配置这个那么事物日志会默认存储到dataDir制定的目录，这样会严重影响zk的性能，当zk吞吐量较大的时候，产生的事物日志、快照日志太多

#clientPort：

这个端口就是客户端连接 Zookeeper 服务器的端口，Zookeeper 会监听这个端口，接受客户端的访问请求。修改他的端口改大点

# 创建myid文件

myid文件在快照目录下存放的标识本台服务器的文件，他是整个zookeeper集群用来发现彼此的一个重要标识。

echo "1" > /var/zookeeper/data/myid

# 配置kafka启动参数

[root@dianliangcaiji139 ~]# cd /usr/local/kafka\_2.10-0.10.0.0/config/

[root@dianliangcaiji139 config]# pwd

/usr/local/kafka\_2.10-0.10.0.0/config

[root@dianliangcaiji139 config]# vi server.properties



# 复制拷贝当前kafka实例到其他机器

在A服务器上将/root/lk目录下所有的文件传输到B的/home/lk/cpfile目录下，命令为：scp -r /root/lk [root@43.224.34.73:/home/lk/cpfile](mailto:root@43.224.34.73:/home/lk/cpfile)。

按如上命令将/usr/local/kafka\_2.10-0.10.0.0拷贝到172.16.15.7

机器。

复制/usr/local/kafka\_2.10-0.10.0.0一份到当前机器的/usr/local/kafka\_2.10-0.10.0.0-9093

# 修改其他两个实例的配置文件

1. 修改myid配置文件；
2. 修改同一台机器另一个zookeeper实例里面的dataDir和dataLogDir。
3. 修改server.properties文件里面的broker.id、host.name（内网iP地址）、advertised.listeners（公网IP地址）

# 分别启动zookeeper、kafka

注意先把zookeeper集群起完再起kafka，起zookeeper的时候会报连不上其他zookeeper的错，这错可以忽略，最后zookeeper都起完就不会报错了。

启动zookeeper

/usr/local/kafka\_2.10-0.10.0.0/bin/zookeeper-server-start.sh /usr/local/kafka\_2.10-0.10.0.0/config/zookeeper.properties &

启动kafka，注意如下两行是一条命令

/usr/local/kafka\_2.10-0.10.0.0/bin/kafka-server-start.sh /usr/local/kafka\_2.10-0.10.0.0/config/server.properties &

创建主题

/usr/local/kafka\_2.10-0.10.0.0/bin/kafka-topics.sh --create --zookeeper 192.168.7.100:12181 --replication-factor 2 --partitions 1 --topic shuaige

生产消息

/usr/local/kafka\_2.10-0.10.0.0/bin/kafka-console-producer.sh --broker-list localhost:9092 --topic test This is the first message

消费消息

/usr/local/kafka\_2.10-0.10.0.0/bin/kafka-console-consumer.sh --zookeeper localhost:12181 --topic test --from-beginning