**测试简单实例**

接下来在Ubuntu系统环境下测试简单的实例。Mac系统请自己按照安装的位置，切换到相应的指令。按顺序执行如下命令：

1. # 进入kafka所在的目录
2. cd /usr/local/kafka
3. bin/zookeeper-server-start.sh config/zookeeper.properties

Shell 命令

命令执行后不会返回Shell命令输入状态,zookeeper就会按照默认的配置文件启动服务,请千万不要关闭当前终端.启动新的终端，输入如下命令：

1. cd /usr/local/kafka
2. bin/kafka-server-start.sh config/server.properties

Shell 命令

kafka服务端就启动了,请千万不要关闭当前终端。启动另外一个终端,输入如下命令:

1. cd /usr/local/kafka
2. bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic dblab

Shell 命令

topic是发布消息发布的category,以单节点的配置创建了一个叫dblab的topic.可以用list列出所有创建的topics,来查看刚才创建的主题是否存在。

1. bin/kafka-topics.sh --list --zookeeper localhost:2181

Shell 命令

可以在结果中查看到dblab这个topic存在。接下来用producer生产点数据：

1. bin/kafka-console-producer.sh --broker-list localhost:9092 --topic dblab

Shell 命令

并尝试输入如下信息：

hello hadoop

hello xmu

hadoop world

然后再次开启新的终端或者直接按CTRL+C退出。然后使用consumer来接收数据,输入如下命令：

1. cd /usr/local/kafka
2. bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic dblab --from-beginning

Shell 命令

便可以看到刚才产生的三条信息。说明kafka安装成功。