# KafKa基本入门教程（windows安裝）

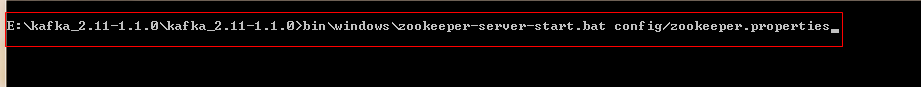
<https://archive.apache.org/dist/kafka/1.1.0/kafka_2.11-1.1.0.tgz>

本教程开始前，假设你已经下载并解压到相应目录下

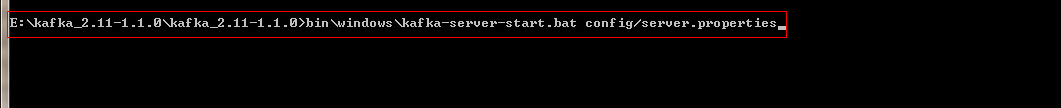
windows 下启动kafka

## 第1步：启动服务器

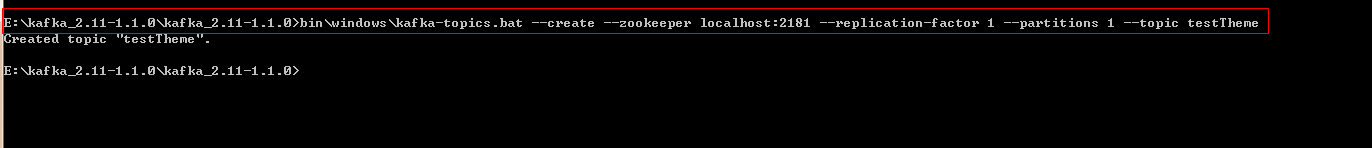
进入到kafka安装目录  
1.先启动zookeeper  
bin\windows\zookeeper-server-start.bat config/zookeeper.properties



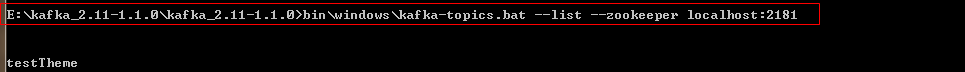
2.再启动kafka  
bin\windows\kafka-server-start.bat config/server.properties



第2步：创建一个主题  
bin\windows\kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic testTheme

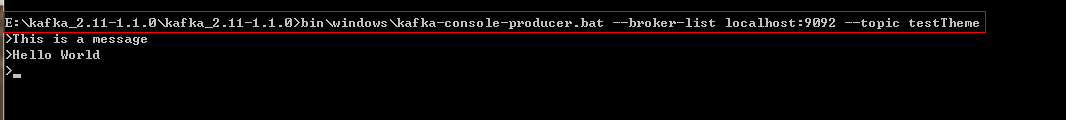


如果我们运行list topic命令，我们现在可以看到该主题：  
bin\windows\kafka-topics.bat --list --zookeeper localhost:2181



第3步：发送一些消息  
Kafka带有一个命令行客户端，它将从文件或标准输入中获取输入，并将其作为消息发送到Kafka集群。默认情况下，每行将作为单独的消息发送。运行生产者，然后在控制台中输入几条消息发送到服务器。  
bin\windows\kafka-console-producer.bat --broker-list localhost:9092 --topic testTheme

This is a message  
Hello World

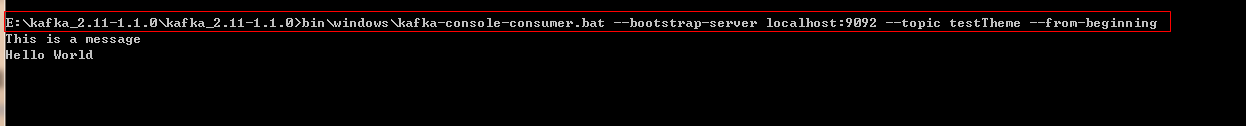


第4步：启动消费者

新版本：  
bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic testTheme --from-beginning

老版本：

bin\windows\kafka-console-consumer.bat --zookeeper localhost:9092 --topic testTheme --from-beginning

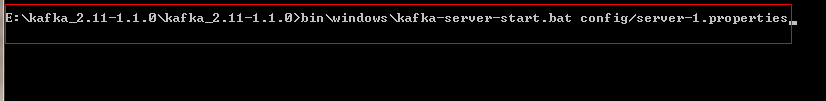


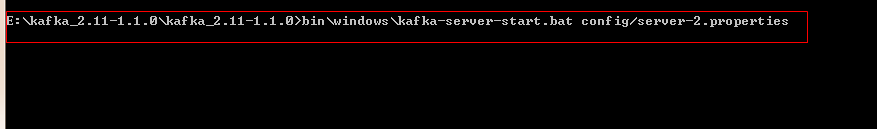
第5步：设置多代理群集  
首先我们为每个代理创建一个配置文件

server-1.properties  
server-2.properties

现在编辑这些新文件并设置下列属性：  
server-1.properties:  
    broker.id=1  
    listeners=PLAINTEXT://:9093  
    log.dir=/tmp/kafka-logs-1  
   
server-2.properties:  
    broker.id=2  
    listeners=PLAINTEXT://:9094  
    log.dir=/tmp/kafka-logs-2  
该broker.id属性是群集中每个节点的唯一且永久的名称。我们必须重写端口和日志目录，因为我们在同一台机器上运行这些端口和日志目录，并且我们希望让所有代理都试图在同一个端口上注册或覆盖彼此的数据。

我们已经有Zookeeper和我们的单节点了，所以我们只需要启动两个新节点：  
bin\windows\kafka-server-start.bat config/server-1.properties

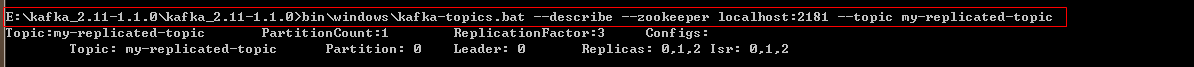
  
bin\windows\kafka-server-start.bat config/server-2.properties



现在创建一个复制因子为3的新主题：  
bin\windows\kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 3 --partitions 1 --topic my-replicated-topic

IMG_264

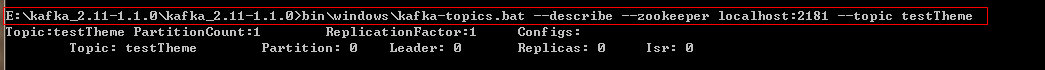
现在我们有一个集群，我们怎么知道哪个经纪人在做什么？要查看运行“描述主题”命令：  
bin\windows\kafka-topics.bat --describe --zookeeper localhost:2181 --topic my-replicated-topic  
Topic:my-replicated-topic   PartitionCount:1    ReplicationFactor:3 Configs:  
    Topic: my-replicated-topic  Partition: 0    Leader: 1   Replicas: 1,2,0 Isr: 1,2,0



这里做一下解释。第一行给出了所有分区的摘要，每个附加行提供了有关一个分区的信息。由于我们只有一个分区，所以只有一行。  
“Leader(leader)”是负责给定分区的所有读写操作的节点。每个节点将成为分区随机选择部分的领导者。  
“Replicas(副本)”是复制此分区的日志的节点列表，无论他们是领导者还是他们现在都活着。  
“Isr”是一组“同步”副本。这是复制品列表的子集，目前活着并被引导到领导者身边。

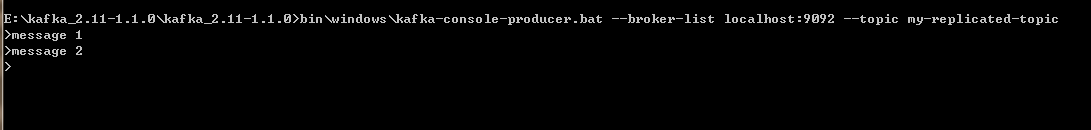
请注意，在我的示例中，节点1是该主题唯一分区的领导者。

我们可以在我们创建的原始主题上运行相同的命令，以查看它的位置：  
bin\windows\kafka-topics.bat --describe --zookeeper localhost:2181 --topic testTheme  
Topic:testTheme   PartitionCount:1    ReplicationFactor:1 Configs:  
    Topic: testTheme  Partition: 0    Leader: 0   Replicas: 0 Isr: 0

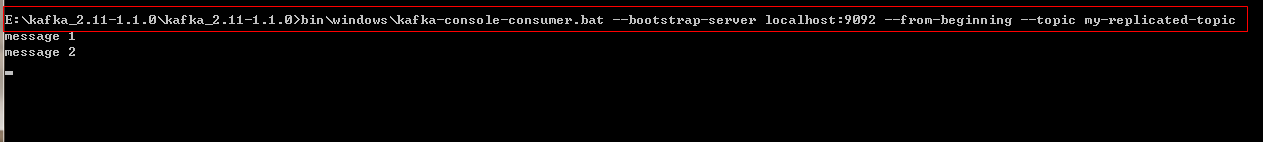


所以在这里并不奇怪,原始主题没有副本，并且在服务器0上，它是我们创建群集时唯一的服务器。

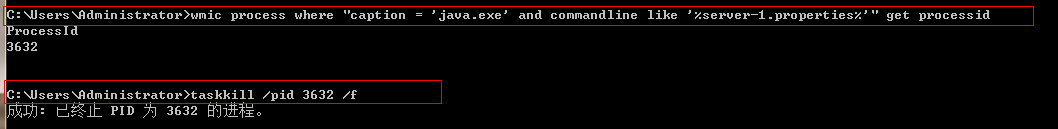
让我们发布一些消息给我们的新主题：  
bin\windows\kafka-console-producer.bat --broker-list localhost:9092 --topic my-replicated-topic  
message 1  
message 2



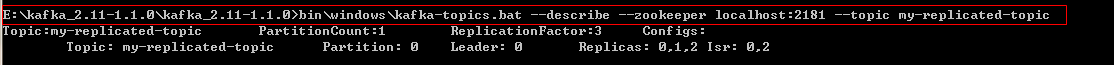
让我们获取这些消息:  
bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --topic my-replicated-topic  
message 1  
message 2



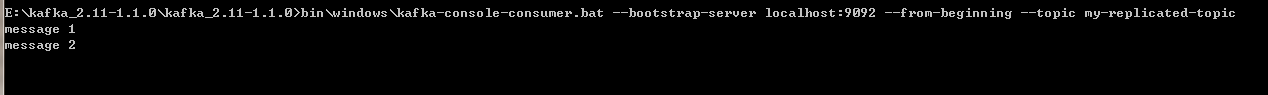
现在我们来测试容错。由于leader1是领导者，现在我们关闭它  
wmic process where "caption = 'java.exe' and commandline like '%server-1.properties%'" get proce  
ProcessId  
3632  
taskkill /pid 3632 /f



leader已切换到其中一个从属节点，并且节点1不再处于同步副本集中：  
bin\windows\kafka-topics.bat --describe --zookeeper localhost:2181 --topic my-replicated-topic  
Topic:my-replicated-topic   PartitionCount:1    ReplicationFactor:3 Configs:  
    Topic: my-replicated-topic  Partition: 0    Leader: 0   Replicas: 0,1,2 Isr: 0,2



但是即使原先的leadern停掉，这些消息仍然可用于消费:  
bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --topic my-replicated-topic  
message 1  
message 2



入门教程到这里就结束了，是不是非常简单，Linux上的操作和windows一样只需要

把bin\windows 换成bin/   .bat换成.sh即可。

点击此处直接下载：[kafka下载](http://mirror.bit.edu.cn/apache/kafka/1.1.0/kafka_2.11-1.1.0.tgz)

官网教程：<http://kafka.apache.org/quickstart>