Linux Kafka配合zookeeper启动并整合SpringBoot

1. 下载kafka，我下载的是kafka_2.12-1.0.0.tgz，所以pom添加依赖的时候也要添加对应的jar包
2. 下载zookeeper
3. 解压，首先启动zookeeper，修改zookeeper/conf下面的zoo_sample.cfg, 修改里面的 `dataDir=/opt/zkdata`(举例)，将这个文件改名为zoo.cfg或者复制出一个名为zoo.cfg的文件。
4. 启动zookeeper `./bin/zkServer.sh start`
5. 修改kafka/config 的server.properties文件，修改 `log.dirs=/opt/kadata` `advertised.listeners=PLAINTEXT://192.168.113.136:9092`(这一步很重要，暴露出地址外网能访问，主机才能访问虚拟机)
6. 启动kafka

   `./bin/kafka-server-start.sh -daemon config/server.properties` 一定要加上 -daemon，加上意味着程序在后台运行
7. 创建一个名为test 的topic

   `./kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic test`
8. 现在添加pom依赖，一定要根据自己下的kafka下载相应版本

```
<dependency>
    <groupId>org.apache.kafka</groupId>
    <artifactId>kafka_2.12</artifactId>
    <version>1.0.0</version>
</dependency>
<dependency>
    <groupId>org.springframework.kafka</groupId>
    <artifactId>spring-kafka</artifactId>
    <version>2.1.9.RELEASE</version>
</dependency>
```

9. 在application.properties添加一些配置

```
spring.kafka.bootstrap-servers=192.168.113.136:9092
spring.kafka.producer.key-
serializer=org.apache.kafka.common.serialization.StringSerializer
spring.kafka.producer.value-
serializer=org.apache.kafka.common.serialization.StringSerializer
spring.kafka.producer.batch-size=4096
spring.kafka.producer.buffer-memory=40960
spring.kafka.consumer.group-id=test
spring.kafka.consumer.enable-auto-commit=true
spring.kafka.consumer.auto-commit-interval=100
spring.kafka.consumer.key-
serializer=org.apache.kafka.common.serialization.StringDeserializer
spring.kafka.consumer.value-
serializer=org.apache.kafka.common.serialization.StringDeserializer
spring.kafka.consumer.bootstrap-servers=192.168.113.136:9092
spring.kafka.consumer.auto-offset-reset=earliest
```

10. 编写一个ProducerController

```
package com.niu.springboot.formal;

import com.niu.springboot.kafka.Response;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.kafka.core.KafkaTemplate;
import org.springframework.web.bind.annotation.*;

/**
 * @Auth justinniu
 * @Date 2018/9/3
 * @Desc
 */

@RestController
@RequestMapping("/kafka")
public class ProducerController {
    protected final Logger logger = LoggerFactory.getLogger(this.getClass());
    @Autowired
    private KafkaTemplate<String, String> kafkaTemplate;

    @RequestMapping(value = "/send", method = RequestMethod.GET)
    public String sendKafka(@RequestParam("message") String message) {
        try {
            logger.info("kafka的消息={}", message);
            kafkaTemplate.send("test", "key", message);
            logger.info("发送kafka成功.");
            return "successs";
        } catch (Exception e) {
            logger.error("发送kafka失败", e);
            return "failure";
        }
```

```
    }

}
```

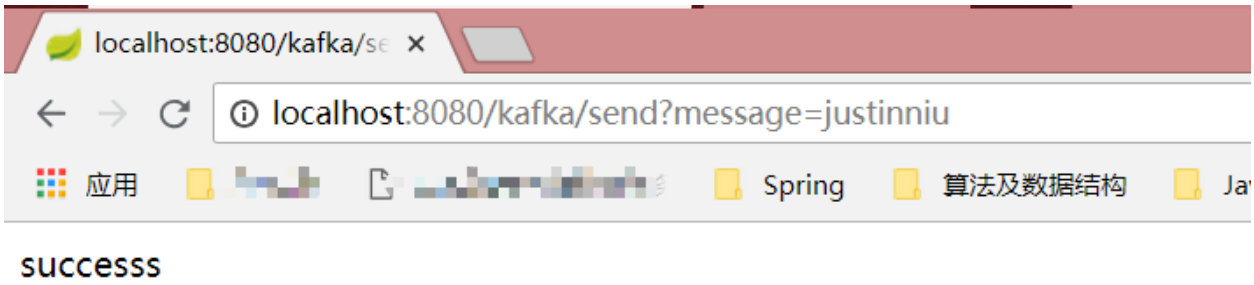
11. 编写一个ConsumerController

```
package com.niu.springboot.formal;

import org.apache.kafka.clients.consumer.ConsumerRecord;
import org.springframework.kafka.annotation.KafkaListener;
import org.springframework.stereotype.Component;

/**
 * @Auth justinniu
 * @Date 2018/9/3
 * @Desc
 */
@Component
public class TestConsumer {
    @KafkaListener(topics = "test")
    public void listen (ConsumerRecord<?, ?> record) throws Exception {
        System.out.printf("topic = %s, offset = %s, value = %s \n", record.topic(),
record.key(), record.value());
    }
}
```

12. 测试结果




```
2018-09-04 10:25:01.713  INFO 5776 --- [ntainer#0-0-C-1] o.a.k.c.c.internals.AbstractCoordinator  : [Consumer clientId=consumer-1, groupId=test] Discovered group coordinator 192.168.113.136:9092 (id: 2147483647 rack: null)
2018-09-04 10:25:01.715  INFO 5776 --- [ntainer#0-0-C-1] o.a.k.c.c.internals.ConsumerCoordinator  : [Consumer clientId=consumer-1, groupId=test] Revoking previously assigned partitions []
2018-09-04 10:25:01.716  INFO 5776 --- [ntainer#0-0-C-1] o.s.k.l.KafkaMessageListenerContainer    : partitions revoked: []
2018-09-04 10:25:01.716  INFO 5776 --- [ntainer#0-0-C-1] o.a.k.c.c.internals.AbstractCoordinator  : [Consumer clientId=consumer-1, groupId=test] (Re-)joining group
2018-09-04 10:25:05.199  INFO 5776 --- [ntainer#0-0-C-1] o.a.k.c.c.internals.AbstractCoordinator  : [Consumer clientId=consumer-1, groupId=test] Successfully joined group with generation 4
2018-09-04 10:25:05.201  INFO 5776 --- [ntainer#0-0-C-1] o.a.k.c.c.internals.ConsumerCoordinator  : [Consumer clientId=consumer-1, groupId=test] Setting newly assigned partitions [test-0]
2018-09-04 10:25:05.205  INFO 5776 --- [ntainer#0-0-C-1] o.s.k.l.KafkaMessageListenerContainer    : partitions assigned: [test-0]
2018-09-04 10:25:19.661  INFO 5776 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring FrameworkServlet 'dispatcherServlet'
2018-09-04 10:25:19.661  INFO 5776 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet        : FrameworkServlet 'dispatcherServlet': initialization started
2018-09-04 10:25:19.681  INFO 5776 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet        : FrameworkServlet 'dispatcherServlet': initialization completed in 20 ms
2018-09-04 10:25:19.709  INFO 5776 --- [nio-8080-exec-1] c.n.springboot.formal.CollectController  : kafka的消息=justinniu
2018-09-04 10:25:19.713  INFO 5776 --- [nio-8080-exec-1] o.a.k.clients.producer.ProducerConfig    : ProducerConfig values:
```

```
2018-09-04 10:25:19.727  INFO 5776 --- [nio-8080-exec-1] o.a.kafka.common.utils.AppInfoParser     : Kafka version : 1.0.1
2018-09-04 10:25:19.727  INFO 5776 --- [nio-8080-exec-1] o.a.kafka.common.utils.AppInfoParser     : Kafka commitId : c0518aa65f25317e
2018-09-04 10:25:19.747  INFO 5776 --- [nio-8080-exec-1] c.n.springboot.formal.CollectController  : 发送kafka成功.
topic = test, offset = key, value = justinniu
```

```
[root@localhost kafka]# ./bin/kafka-console-consumer.sh --zookeeper localhost:2181 --from-beginning --topic test
OpenJDK 64-Bit Server VM warning: If the number of processors is expected to increase from one, then you should configure the number of parallel GC threads appropriately using
arallelGCThreads=N
Using the ConsoleConsumer with old consumer is deprecated and will be removed in a future major release. Consider using the new consumer by passing [bootstrap-server] instead
okeeper].
123
justinniu
justinniu
```