

[首页](#) [资讯](#) [精华](#) [论坛](#) [问答](#) [博客](#) [专栏](#) [群组](#) [更多 ▾](#)[您还未登录！](#) [登录](#) [注册](#)[电子商务网站开发](#) [sql题](#) [高端网站建设](#) [电商网站设计](#) [电脑编程入门自学](#) [apache入门](#) [scala api](#)[如何注册商贸公司](#) [云计算编程](#) [游戏编程要学什么](#) [企业网站优化](#) [xml文件是什么](#) [大数据课程体系](#) [做网站](#) [大数据学习](#)[一般实木床价格](#) [盛利](#) [企业网站建设](#) [管易erp系统](#) [视频会议系统方案](#) [飞禽走兽大白鲨游戏](#) [python自动化](#) [电商平△](#)

czj4451

- [博客](#)
- [微博](#)
- [相册](#)
- [收藏](#)
- [留言](#)
- [关于我](#)



Kafka 安装和测试

[关闭](#)

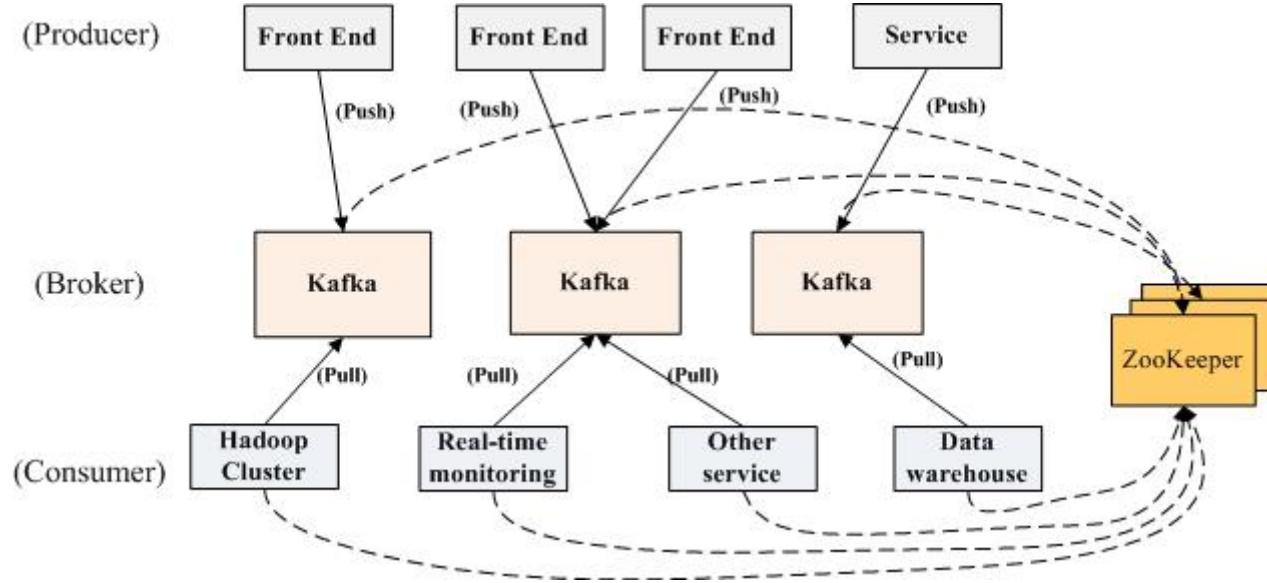
博客分类：

- [middleware](#)

kafkazookeeperscalajvm

1. 简介

kafka（官网地址：<http://kafka.apache.org>）是一款分布式消息发布和订阅的系统，具有高性能和高吞吐率。



- 消息的发布 (publish) 称作producer，消息的订阅 (subscribe) 称作consumer，中间的存储阵列称作broker。
- 多个broker协同合作，producer、consumer和broker三者之间通过zookeeper来协调请求和转发。
- producer产生和推送(push)数据到broker，consumer从broker拉取(pull)数据并进行处理。
- broker端不维护数据的消费状态，提升了性能。
- 直接使用磁盘进行存储，线性读写，速度快：避免了数据在JVM内存和系统内存之间的复制，减少耗性能的创建对象和垃圾回收。
- Kafka使用scala编写，可以运行在JVM上。

2. 安装：

- 首先安装JRE/JDK

[Linux安装JDK](#)

- 下载kafka

进入下载页面：<http://kafka.apache.org/downloads.html>

选择Binary downloads下载 (Source download需要编译才能使用)

也可以直接在linux终端下载：

Shell代码 ☆

```
1. wget -q http://apache.fayea.com/apache-mirror/kafka/0.8.1/kafka_2.8.0-0.8.1.tgz
```

c. 解压

Shell代码 ☆

```
1. tar -xzvf kafka_2.8.0-0.8.1.tgz
2. rm kafka_2.8.0-0.8.1.tgz
3. cd kafka_2.8.0-0.8.1
```

目录：

```
root@ubuntu:/usr/kafka/kafka_2.8.0-0.8.1# ls
bin config libs LICENSE NOTICE
root@ubuntu:/usr/kafka/kafka_2.8.0-0.8.1# cd bin && ls
kafka-console-consumer.sh      kafka-replay-log-producer.sh      kafka-topics.sh
kafka-console-producer.sh      kafka-run-class.sh          windows
kafka-consumer-perf-test.sh    kafka-server-start.sh      zookeeper-server-start.sh
kafka-preferred-replica-election.sh  kafka-server-stop.sh      zookeeper-server-stop.sh
kafka-producer-perf-test.sh    kafka-simple-consumer-perf-test.sh  zookeeper-shell.sh
kafka-reassign-partitions.sh   kafka-simple-consumer-shell.sh
root@ubuntu:/usr/kafka/kafka_2.8.0-0.8.1/bin# cd ../config && ls
consumer.properties  producer.properties  test-log4j.properties  zookeeper.properties
log4j.properties      server.properties    tools-log4j.properties
root@ubuntu:/usr/kafka/kafka_2.8.0-0.8.1/config# cd ../libs && ls
jopt-simple-3.2.jar        metrics-annotation-2.2.0.jar  snappy-java-1.0.5.jar
kafka_2.8.0-0.8.1.jar     metrics-core-2.2.0.jar      zkclient-0.3.jar
kafka_2.8.0-0.8.1.jar.asc  scala-library-2.8.0.jar    zookeeper-3.3.4.jar
log4j-1.2.15.jar          slf4j-api-1.7.2.jar
```

关闭

/bin 启动和停止命令等。

/config 配置文件

/libs 类库

d. 修改配置

Kafka默认开启JVM压缩指针，但只是在64位的HotSpot VM受支持，如果安装了32位的HotSpot VM，需要修改**/bin/kafka-run-class.sh**文件

Shell代码 

1. vi bin/kafka-run-class.sh

找到如下行：

Sh代码 

1. KAFKA_JVM_PERFORMANCE_OPTS="-server -XX:+UseCompressedOops -XX:+UseParNewGC -XX:+UseConcMarkSweepGC -XX:+CMSClassUnloadingEnabled -XX:+CMSScavengeBeforeRemark -XX:+DisableExplicitGC -Djava.awt.headless=true"

去除~~-XX:+UseCompressedOops~~参数

3. 启动和停止

[关闭](#)

启动Zookeeper server：

Shell代码 

1. bin/zookeeper-server-start.sh config/zookeeper.properties &

&是为了能退出命令行

启动Kafka server:

Shell代码 

1. bin/kafka-server-start.sh config/server.properties &

停止Kafka server

Shell代码 

1. bin/kafka-server-stop.sh

停止Zookeeper server:

Shell代码 

1. bin/zookeeper-server-stop.sh

4. 单机连通性测试

运行producer :

Shell代码 

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

早版本的Kafka， --broker-list localhost:9092需改为--zookeeper localhost:2181

关闭

运行consumer :

Shell代码 

1. bin/kafka-console-consumer.sh --zookeeper localhost:**2181** --topic test --from-beginning

在producer端输入字符串并回车， 查看consumer端是否显示。

5. 分布式连通性测试

Zookeeper Server, Kafka Server, Producer都放在服务器server1上， ip地址为192.168.1.10

Consumer放在服务器server2上， ip地址为192.168.1.12。

分别运行server1的producer和server2的consumer，

Shell代码 

1. bin/kafka-console-producer.sh --broker-list 192.168.1.10:9092 --topic test

Shell代码 

1. bin/kafka-console-consumer.sh --zookeeper 192.168.1.10:2181 --topic test --from-beginning

在producer的console端输入字符串， consumer报Connection refused错误：

```
kafka.common.KafkaException: fetching topic metadata for topics [Set(test)] from
broker [ArrayBuffer(id:0,host:ubuntu,port:9092)] failed
  at kafka.client.ClientUtils$.fetchTopicMetadata(ClientUtils.scala:67)
  at kafka.client.ClientUtils$.fetchTopicMetadata(ClientUtils.scala:88)
  at kafka.consumer.ConsumerFetcherManager$LeaderFinderThread.doWork(ConsumerFetcherManager.scala:66)
  at kafka.utils.ShutdownableThread.run(ShutdownableThread.scala:51)
Caused by: java.net.ConnectException: Connection refused
  at sun.nio.ch.Net.connect0(Native Method)
  at sun.nio.ch.Net.connect(Net.java:465)
  at sun.nio.ch.Net.connect(Net.java:457)
  at sun.nio.ch.SocketChannelImpl.connect(SocketChannelImpl.java:670)
  at kafka.network.BlockingChannel.connect(BlockingChannel.scala:57)
  at kafka.producer.SyncProducer.connect(SyncProducer.scala:141)
  at kafka.producer.SyncProducer.getOrMakeConnection(SyncProducer.scala:15
6)
  at kafka.producer.SyncProducer.kafka$producer$SyncProducer$$doSend(SyncP
roducer.scala:68)
  at kafka.producer.SyncProducer.send(SyncProducer.scala:112)
  at kafka.client.ClientUtils$.fetchTopicMetadata(ClientUtils.scala:53)
  ... 3 more
```

关闭

broker, producer和consumer都注册到zookeeper上， producer和consumer的参数明确指定。问题出在broker的配置文件server.properties上：

Properties代码



1. # Hostname the broker will bind to. If not set, the server will bind to all interfaces
2. #host.name=localhost

host名称没有指定，就是127.0.0.1，consumer去broker拿数据就有问题。设置为192.168.1.10，重启服务就好了。

- [查看图片附件](#)

分享到：

[NIO UDP 编程 | Spring AOP系列之五：后置通知](#)

- 2014-04-15 10:31
- 浏览 89470
- [评论\(1\)](#)
- 分类:[企业架构](#)
- [相关推荐](#)

关闭

参考知识库



[语音识别与合成知识库](#) 358 关注 | 263 收录



[计算机视觉知识库](#) 895 关注 | 248 收录



[自然语言理解和处理知识库](#) 338 关注 | 87 收录



[知识工程知识库](#) 105 关注 | 69 收录

评论

1 楼 [無名2](#) 2016-04-14

<http://orchome.com/kafka/index>

这篇文章写得不错呀！！！

发表评论



[您还没有登录,请您登录后再发表评论](#)



czj4451

关闭

- 浏览: 626588 次
- 性别: ♂
- 来自: 杭州
- 我现在离线

最近访客

[更多访客>>](#)

[usagoole](#)[caoliangmin](#)[772191140](#)[qupengdl](#)

文章分类

- [全部博客 \(191\)](#)
- [ant/maven \(6\)](#)
- [algorithm \(5\)](#)
- [tomcat/weblogic/jboss \(6\)](#)
- [javascript/jquery \(13\)](#)
- [java \(33\)](#)
- [flex/flash \(0\)](#)
- [JPA/Hibernate/myBatis \(18\)](#)
- [java concurrent \(7\)](#)
- [test \(2\)](#)
- [windows/linux \(6\)](#)
- [java collection \(7\)](#)
- [design pattern \(2\)](#)
- [life/health \(3\)](#)
- [database \(12\)](#)
- [IDE \(4\)](#)
- [spring/ejb \(20\)](#)
- [html/css/ckeditor \(7\)](#)
- [jsp/servlet \(3\)](#)
- [java io \(13\)](#)
- [java security \(4\)](#)

关闭

- [ini \(0\)](#)
- [svn/git \(2\)](#)
- [english \(2\)](#)
- [java jmx \(1\)](#)
- [xml \(1\)](#)
- [struts/springmvc \(9\)](#)
- [middleware \(1\)](#)
- [cache \(1\)](#)
- [cglib \(3\)](#)

社区版块

- [我的资讯 \(0\)](#)
- [我的论坛 \(0\)](#)
- [我的问答 \(0\)](#)

存档分类

- [2014-11 \(1\)](#)
- [2014-10 \(1\)](#)
- [2014-09 \(1\)](#)
- [更多存档...](#)

最新评论

- [lwj1113](#) : 谢谢博主这么细致的demo;在系列五中通过testng测试类跑 ...
[myBatis系列之五：与Spring3集成](#)
- [lidaxue](#) : mark!
[logback系列之四：输出日志到不同文件](#)
- [陌上君](#) : 引用[lis[color=indigo][/color]t][...
[myBatis系列之七：事务管理](#)
- [Hibernate1](#) : 讲解的很不错，收益了。感谢发表者；
[Hibernate 多对一关联配置](#)

关闭

- [santouren](#) : 感谢整理, Java初学者受益良多

[Thread 源码分析](#)

声明 : ITeye文章版权属于作者, 受法律保护。没有作者书面许可不得转载。若作者同意转载, 必须以超链接形式标明文章原始出处和作者。

**全国30座城市
140个甲级写字楼
办公室任你选。**



雷格斯服务式办公室