

kafka集群部署

1、zookeeper安装

分布式协调服务

1、第一步：下载，解压，复制zoo.cfg，修改zoo.cfg

```
1 dataDir=/data/zookeeper/data
2 dataLogDir=/data/zookeeper/logs
3
4 server.1=192.168.18.22:8880:7770
5 server.2=192.168.18.24:8880:7770
6 server.3=192.168.18.27:8880:7770
```

2、复制到另外两个节点上

3、分别在三个节点的/data/zookeeper/data 目录下创建myid文件，内容为：

第一个节点为 1，第二个节点为 2，第三个节点为 3。

4、启动

```
1 ./bin/zkServer.sh start
```

5、确定启动是否正常

```
1 ./bin/zkServer.sh status
```

2、kafka的安装

1、修改 server.properties 文件

```
1 broker.id=0
2 listeners=PLAINTEXT://192.168.18.22:9092
3 log.dirs=/data/kafka/kafka-logs
4 zookeeper.connect=192.168.18.22:2181,192.168.18.24:2181,192.168.18.27:2181
```

```
1 broker.id=1
2 listeners=PLAINTEXT://192.168.18.24:9092
3 log.dirs=/data/kafka/kafka-logs
4 zookeeper.connect=192.168.18.22:2181,192.168.18.24:2181,192.168.18.27:2181
```

```
1 broker.id=2
2 listeners=PLAINTEXT://192.168.18.27:9092
3 log.dirs=/data/kafka/kafka-logs
4 zookeeper.connect=192.168.18.22:2181,192.168.18.24:2181,192.168.18.27:2181
```

2、启动

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

3、创建topic（默认需要手动创建，可以修改配置文件让kafka自动创建不存在的topic）

创建一个名称为 test topic，指定 test topic有一个分区，一个副本。

```
1 | ./bin/kafka-topics.sh --create --topic test --bootstrap-server  
192.168.18.22:9092 --partitions 1 --replication-factor 1
```

查看topic信息

```
1 | ./bin/kafka-topics.sh --describe --topic test --bootstrap-server  
192.168.18.22:9092
```

```
[root@master kafka_2.13-2.8.0]# ./bin/kafka-topics.sh --describe --topic test --bootstrap-server 192.168.18.22:9092  
Topic: test      TopicId: E19epu3WQ0ael2PcxAuSrw PartitionCount: 1      ReplicationFactor: 1      Configs: segment.bytes=1073741824  
    Topic: test      Partition: 0      Leader: 0      Replicas: 0      Isr: 0
```

4、发送消息

用kafka自带的命名模拟消息发送

```
1 | ./bin/kafka-console-producer.sh --bootstrap-server 192.168.18.22:9092 --topic  
test
```

5、消费消息

用kafka自带的命名模拟消息消费

```
1 | ./bin/kafka-console-consumer.sh --bootstrap-server 192.168.18.22:9092 --topic  
test --from-beginning
```

3、kafka用在上面场景下

1、能不能替代传统的消息消息中间件??

activemq rabbitmq 比较

消息低延迟（端到端），高吞吐，并且提供了强大的高可用性保证。

2、网站信息收集

3、zipkin skywakling

4、收集

5、流处理系统

6、文件传输