☰

600 days ago in   LINUX   MESSAGING   APACHE KAFKA   ZOOKEEPER

Installing Apache Kafka Multi nodes and Multi Brokers on Ubuntu 🐦

Introduction

In the previous article I have shown you how to install **Apache Kafka** with **Single Node** and **Multi brokers**. However In the really systems, will have very many requests from clients to our system, so we should have many kafka brokers locate in the other server to handler many messages (Distributed Systems).

Because of that, In the **production** environments of distributed systems, we will need to install Apache Kafka with **Multi nodes** and **Multi brokers** on the others server.

In this article, I have shown to you how to install **Apache Kafka** with **Multi Nodes** and **Multi Brokers** on the same server.

Installation

1) Prerequisites

Before install Kafka, we will need to have:

- Ubuntu Server 14.04
- Java 7 or later.
- In this article, I will install and configure Kafka and Zookeeper on 3 Servers (or n+1 servers, with n is an even number). 3 Servers has internal IP:

- Server 1: 192.168.1.86

- Server 2: 192.168.1.88

- Server 3: 192.168.1.89

- All of the steps will process on all three servers.

2) Install Java

Install Java in this case is the same with **single node** and **single broker**. Please see it in:

- https://cuongba.com/installing-apache-kafka-single-node-and-single-broker-on-ubuntu/

3) Install Apache Zookeeper

Download Zookeeper on all three servers:

```
$ wget http://mirrors.maychuviet.vn/apache/zookeeper/stable/zookeeper-3.4.6.tar.gz
$ tar -xzvf zookeeper-3.4.6.tar.gz
$ cd zookeeper-3.4.6
$ cp conf/zoo_sample.cfg conf/zoo.cfg
```

configuration for Zookeeper:

```
tickTime=2000
initLimit=10
syncLimit=5
dataDir=/var/zookeeper/data # <--- Important
clientPort=2181
```

```
maxClientCnxns =60
autopurge .snapRetainCount =3
autopurge .purgeInterval =1
server .1=192.168 .1.86:2888:3888 # <--- Important
server .2=192.168 .1.88:2888:3888 # <--- Important
server .3=192.168 .1.89:2888:3888 # <--- Important
```

If you have more 3 servers, you can continue append server configuration.

Create the zookeeper unique identifiers on all the nodes:

```
ngoccuong@server1 :/var# echo "1" > / var/zookeeper/data/myid
ngoccuong@server2 :/var# echo "2" > / var/zookeeper/data/myid
ngoccuong@server3 :/var# echo "3" > / var/zookeeper/data/myid
```

Then, you can run Zookeeper on each server:

```
$ ./bin/zkServer .sh start
```

When you run, will have a Zookeeper is voted to become a master node.

3) Install Apach e Kafka with Multi-Br okers on each n ode
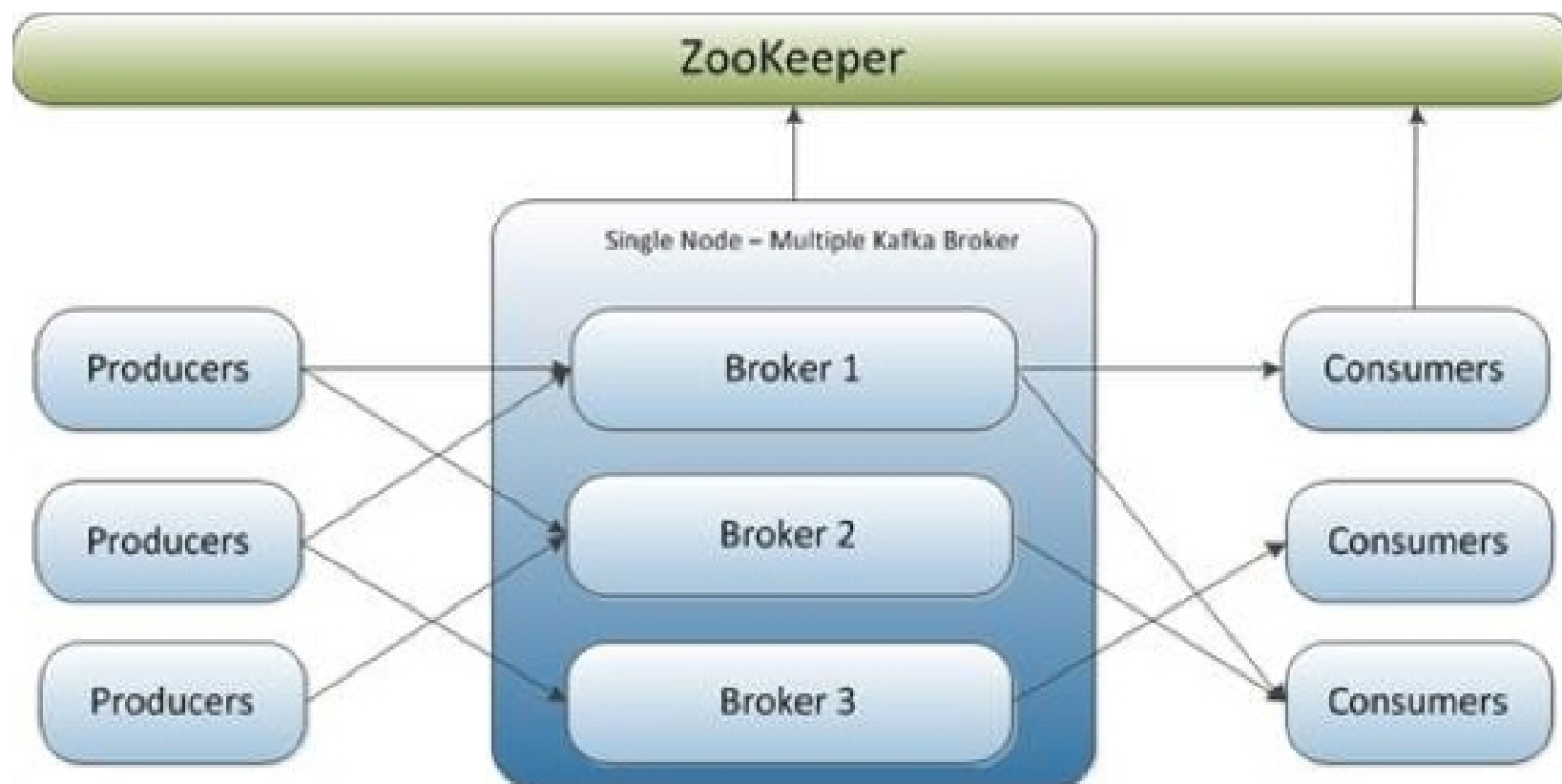
```
# Download Kafka
$ wget http://mirrors.maychuviet.vn/apache//kafka/0.8.2.1/kafka_2.10-0.8.2.1.tgz

# Extra
$ tar -xzvf kafka_2.10-0.8.2.1.tgz
$ cd kafka_2.10-0.8.2.1
```

Now, you want to run how many brokers, simple you just need to create or copy Kafka Config file (config/server.properties). And you should run with the odd number of kafka brokers as 1, 3, 5 ... Each property file defines different values for the following properties:

☰

- **broker.id**
- **port**
- **log.dir**
- **host.name**
- **zookeeper.connect**

To me, I will configure to run 3 brokers:

```
$ cp config/server.properties config/server_1.properties
$ cp config/server.properties config/server_2.properties
```

With **server.properties**:

```
broker.id=0
port=9092
log.dir=/tmp/kafka0-logs
host.name=192.168.1.86
zookeeper.connect=192.168.1.86:2181,192.168.1.88:2181,192.168.1.89:2181
```

With **server_1.properties**:

```
broker.id=1
port=9093
log.dir=/tmp/kafka1-logs
host.name=192.168.1.88
zookeeper.connect=192.168.1.86:2181,192.168.1.88:2181,192.168.1.89:2181
```

With **server_2.properties**:

```
broker.id=2
port=9094
log.dir=/tmp/kafka2-logs
```

≡

```
host.name=192.168.1.89
zookeeper.connect=192.168.1.86:2181,192.168.1.88:2181,192.168.1.89:2181
```

Finally, we will run 3 brokers:

```
$ sudo ./bin/kafka-server-start.sh -daemon config/server.properties
$ sudo ./bin/kafka-server-start.sh -daemon config/server_1.properties
$ sudo ./bin/kafka-server-start.sh -daemon config/server_2.properties
```

Run with **nohup** command, brokers will run in background, and you do not have to open other terminals.

4) Testing

To test Apache Kafka is running, you can use **kafka-console-producer** and **kafka-console-consumer** to send and receive data.

Before, we need to create a topic in Kafka.

```
# Create topic
$ bin/kafka-topics.sh  --create --zookeeper localhost:2181 --replication-factor 1 \
--partition 1 --topic kafkatopic

#List topics in kafka
$ bin/kafka-topics.sh --list --zookeeper localhost:2181
```

Start a consumer to consuming messages.

```
$ bin/kafka-console-consumer.sh --zookeeper localhost:2181 --topic kafkatopic \
--from-beginning
```

Start a producer to sending messages in a new terminal window.

```
$ bin/kafka-console-producer.sh --broker-list localhost:9092 --topic kafkatopic
[2015-09-06 7:26:58,179] WARN Property topic is not valid (kafka.utils.VerifiableProperties )
```

```
Hello Cuong.
This is test message.
```

And should be in the consumer terminal, you will see:

```
$ bin/kafka-console-consumer.sh --zookeeper localhost :2181 --topic kafkatopic \
--from-beginning
Hello Cuong.
This is test message.
```

So, We have installed successful Apache Kafka in single node and Multi broker. **In the next article**, I will show to you **how to install Apache Kafka Multi-Nodes and Multi-Brokers**.

6 Comments   **cuongba.com**   ① **Login** ▾

♡ **Recommend**   ⬆ **Share**   **Sort by Best** ▾

            Join the discussion…

bram • 4 months ago
in this guide are server.properties in server1, server_2.properties in server and server_3.properties in server3,or they are in all server?
⌃ | ⌄ • Reply • Share ›

Gali Jagan • a year ago
when I want to consume messages or to create a topic in multi zookeeper mode then which zook ip address do I need to mention in command
⌃ | ⌄ • Reply • Share ›

Ido Nadler • a year ago
Can you please explain why to use an odd number of Kafka brokers
⌃ | ⌄ • Reply • Share ›

　　Cuong Ba Mod ➔ Ido Nadler • a year ago
　　Hello Ido Nadler,

We should to use an odd number of Kafka brokers because in Kafka, broker leader is elected. If a broker that is a leader does down, Kafka will automatically elect a new broker leader.

∧ | ∨ • Reply • Share ›

Ido Nadler ➜ Cuong Ba • a year ago

Right.

But, correct me if I am wrong, since Kafka only try to balance the leaders and doesn't use a majority algorithm for the election, it can work also with an even number of brokers.

Having an odd number of brokers does make sense, but I couldn't find any reference for it in Kafka docs nor everywere else.

∧ | ∨ • Reply • Share ›

Cuong Ba Mod ➜ Ido Nadler • a year ago

Yes, It can wok with an even number of brokers. So I just notice that "we should" and not "we must" :D. I think we should keep balance is better for Kafka

∧ | ∨ • Reply • Share ›

✉ **Subscribe**   ⒟ **Add Disqus to your siteAdd DisqusAdd**   🔒 **Privacy**