**Zookeeper安装与配置**

# 环境准备

（3台服务器）

192.168.7.100 server1

192.168.7.101 server2

192.168.7.107 server3

1、Linux服务器一台、三台、五台、（2\*n+1），Zookeeper集群的工作是超过半数才能对外提供服务，3台中超过两台超过半数，允许1台挂掉 ，是否可以用偶数，其实没必要。

如果有四台那么挂掉一台还剩下三台服务器，如果在挂掉一个就不行了，这里记住是超过半数。

2、Java jdk1.7及以上 zookeeper是用java写的所以他的需要JAVA环境，java是运行在java虚拟机上的

3、Zookeeper的稳定版本Zookeeper 3.4.13版本

# 安装

本章节操作所有节点一样

首先要注意在生产环境中目录结构要定义好，防止在项目过多的时候找不到所需的项目。目录统一放在/opt下面，首先创建Zookeeper项目目录

cd /opt

mkdir zookeeper #项目目录

cd /opt/zookeeper

mkdir zkdata #存放快照日志

mkdir zkdatalog #存放事物日志

下载Zookeeper并解压软件

cd /opt/zookeeper/

wget https://mirrors.tuna.tsinghua.edu.cn/apache/zookeeper/zookeeper-3.4.13/zookeeper-3.4.13.tar.gz

tar -zxvf zookeeper-3.4.13.tar.gz

# 修改配置

## 配置文件修改

本小节操作所有节点一样，最好在一个节点完成，然后远拷贝到其它节点

进入到解压好的目录里面的conf目录中

注意zoo\_sample.cfg这个文件是官方给我们的zookeeper的样板文件，给他复制一份命名为zoo.cfg（是官方指定的文件命名规则）。

cd /opt/zookeeper/zookeeper-3.4.13/conf

# 复制一份配置文件

cp zoo\_sample.cfg zoo.cfg

3台服务器的配置文件一样，如下

tickTime=2000

initLimit=10

syncLimit=5

dataDir=/opt/zookeeper/zkdata

dataLogDir=/opt/zookeeper/zkdatalog

clientPort=12181

server.1=192.168.7.100:12888:13888

server.2=192.168.7.101:12888:13888

server.3=192.168.7.107:12888:13888

#server.1 这个1是服务器的标识也可以是其他的数字，用来标识服务器，这个标识要写到快照目录下面myid文件里

#192.168.7.100为集群里的IP地址，第一个端口是master和slave之间的通信端口，默认是2888，第二个端口是leader选举的端口，集群刚启动的时候选举或者leader挂掉之后进行新的选举的端口默认是3888

**配置文件解释：**

tickTime：这个时间是作为 Zookeeper 服务器之间或客户端与服务器之间维持心跳的时间间隔，也就是每个 tickTime 时间就会发送一个心跳。

initLimit：这个配置项是用来配置 Zookeeper 接受客户端（这里所说的客户端不是用户连接 Zookeeper 服务器的客户端，而是 Zookeeper 服务器集群中连接到 Leader 的 Follower 服务器）初始化连接时最长能忍受多少个心跳时间间隔数。当已经超过 5个心跳的时间（也就是 tickTime）长度后 Zookeeper 服务器还没有收到客户端的返回信息，那么表明这个客户端连接失败。总的时间长度就是 5\*2000=10 秒

syncLimit：这个配置项标识 Leader 与Follower 之间发送消息，请求和应答时间长度，最长不能超过多少个 tickTime 的时间长度，总的时间长度就是5\*2000=10秒

dataDir：快照日志的存储路径

dataLogDir：事物日志的存储路径，如果不配置这个那么事物日志会默认存储到dataDir制定的目录，这样会严重影响zk的性能，当zk吞吐量较大的时候，产生的事物日志、快照日志太多

clientPort：这个端口就是客户端连接 Zookeeper 服务器的端口，Zookeeper 会监听这个端口，接受客户端的访问请求。修改他的端口改大点

## 创建myid文件

各个节点不一样

登录节点1执行

echo "1" > /opt/zookeeper/zkdata/myid

登录节点2执行

echo "2" > /opt/zookeeper/zkdata/myid

登录节点3执行

echo "3" > /opt/zookeeper/zkdata/myid

# 启动服务并查看

1、启动服务 （3台都需要操作）

cd /opt/zookeeper/zookeeper-3.4.13/bin

./zkServer.sh start

2、检查服务状态

通过status就能看到状态：

./zkServer.sh status

JMX enabled by default

Using config: /opt/zookeeper/zookeeper-3.4.13/bin/../conf/zoo.cfg #配置文件

Mode: follower #他是否为领导

zk集群一般只有一个leader，多个follower，主一般是相应客户端的读写请求，而从主同步数据，当主挂掉之后就会从follower里投票选举一个leader出来。

可以用“jps”查看zk的进程，这个是zk的整个工程的main

#执行命令

jps

20348 Jps

4233 QuorumPeerMain

# 其它补充

**重要配置说明**

1、myid文件和server.myid  在快照目录下存放的标识本台服务器的文件，他是整个zk集群用来发现彼此的一个重要标识。

2、zoo.cfg 文件是zookeeper配置文件 在conf目录里。

3、log4j.properties文件是zk的日志输出文件 在conf目录里用java写的程序基本上有个共同点日志都用log4j，来进行管理。

4、zkEnv.sh和zkServer.sh文件

zkServer.sh 主的管理程序文件

zkEnv.sh 是主要配置，zookeeper集群启动时配置环境变量的文件

5、还有一个需要注意

ZooKeeper server **will not remove old snapshots and log files** when using the default configuration (see autopurge below), this is the responsibility of the operator

zookeeper不会主动的清除旧的快照和日志文件

[参考](https://www.cnblogs.com/luotianshuai/p/5206662.html)