Zookeeper安装和配置

（1）配置zk节点的hosts文件：配置3台机器的ip地址和主机名的对应关系。以下以console主机为例，其hosts文件添加下面3行：

114.55.29.246 console

114.55.29.86 log1

114.55.29.241 log2

​

（2）解压安装配置第一台zk

[root@console local]# tar zxf zookeeper-3.4.6.tar.gz

[root@console local]# cd zookeeper-3.4.6

​

创建快照日志存放目录：

[root@console zookeeper-3.4.6]# mkdir -p dataDir

​

创建事务日志存放目录：

[root@console zookeeper-3.4.6]# mkdir dataLogDir

​

【注意】:如果不配置dataLogDir，那么事务日志也会写在dataDir目录中。这样会严重影响zk的性能。因为在zk吞吐量很高的时候，产生的事务日志和快照日志太多。

修改配置文件，添加如下内容：

[root@console zookeeper-3.4.6]# cd conf

[root@console conf]# mv zoo\_sample.cfg zoo.cfg

​

[root@console conf]# vim zoo.cfg

​

# 存放数据文件

dataDir=/usr/local/zookeeper-3.4.6/dataDir

​

# 存放日志文件

dataLogDir=/usr/local/zookeeper-3.4.6/dataLogDir

​

# zookeeper cluster，2888为选举端口，3888为心跳端口

​

server.1=console:2888:3888

server.2=log1:2888:3888

server.3=log2:2888:3888

在我们配置的dataDir指定的目录下面，创建一个myid文件，

​里面内容为一个数字，用来标识当前主机，

​conf/zoo.cfg文件中配置的server.X中X为什么数字，则myid文件中就输入这个数字：

[root@console ~]# echo "1" > /usr/local/zookeeper-3.4.6/dataDir/myid

​

​

（3）远程复制第一台的zk到另外两台上，并修改myid文件为2和3

[root@console local]# scp -rp zookeeper-3.4.6 root@114.55.29.86:/usr/local/

[root@console local]# scp -rp zookeeper-3.4.6 root@114.55.29.241:/usr/local/

4. 启动和关闭zk

在ZooKeeper集群的每个结点上，执行启动ZooKeeper服务的脚本，如下所示：

[root@console bin]# ./zkServer.sh start

[root@log1 bin]# ./zkServer.sh start

[root@log2 bin]# ./zkServer.sh start

​

日志可查询：/usr/local/zookeeper-3.4.6/bin/zookeeper.out

可以通过命令jps查看Zookeeper进程：

​​

​

​停止zk命令：

# /usr/local/zookeeper-3.4.6/bin/zkServer.sh stop

5. 测试zk集群

可以通过ZooKeeper的脚本来查看启动状态，包括集群中各个结点的角色（或是Leader，或是Follower）

[root@console bin]# ./zkServer.sh status

JMX enabled by default

Using config: /usr/local/zookeeper-3.4.6/bin/../conf/zoo.cfg

Mode: follower

​

[root@log1 bin]# ./zkServer.sh status

JMX enabled by default

Using config: /usr/local/zookeeper-3.4.6/bin/../conf/zoo.cfg

Mode: leader

​

[root@log2 bin]# ./zkServer.sh status

JMX enabled by default

Using config: /usr/local/zookeeper-3.4.6/bin/../conf/zoo.cfg

Mode: follower

通过上面状态查询结果可见，log1是集群的Leader，其余的两个结点是Follower。

另外，可以通过客户端脚本，连接到ZooKeeper集群上。对于客户端来说，ZooKeeper是一个整体，连接到ZooKeeper集群实际上感觉在独享整个集群的服务，所以，你可以在任何一个结点上建立到服务集群的连接。

[root@log2 bin]# ./zkCli.sh -server log1:2181

​

Connecting to log1:2181

2016-03-08 14:21:31,502 [myid:] - INFO [main:Environment@100] - Client environment:zookeeper.version=3.4.6-1569965, built on 02/20/2014 09:09 GMT

2016-03-08 14:21:31,505 [myid:] - INFO [main:Environment@100] - Client environment:host.name=log2

2016-03-08 14:21:31,505 [myid:] - INFO [main:Environment@100] - Client environment:java.version=1.7.0\_80

2016-03-08 14:21:31,507 [myid:] - INFO [main:Environment@100] - Client environment:java.vendor=Oracle Corporation

2016-03-08 14:21:31,507 [myid:] - INFO [main:Environment@100] - Client environment:java.home=/usr/java/jdk1.7.0\_80/jre

2016-03-08 14:21:31,507 [myid:] - INFO [main:Environment@100] - Client environment:java.class.path=/usr/local/zookeeper-3.4.6/bin/../build/classes:/usr/local/zookeeper-3.4.6/bin/../build/lib/\*.jar:/usr/local/zookeeper-3.4.6/bin/../lib/slf4j-log4j12-1.6.1.jar:/usr/local/zookeeper-3.4.6/bin/../lib/slf4j-api-1.6.1.jar:/usr/local/zookeeper-3.4.6/bin/../lib/netty-3.7.0.Final.jar:/usr/local/zookeeper-3.4.6/bin/../lib/log4j-1.2.16.jar:/usr/local/zookeeper-3.4.6/bin/../lib/jline-0.9.94.jar:/usr/local/zookeeper-3.4.6/bin/../zookeeper-3.4.6.jar:/usr/local/zookeeper-3.4.6/bin/../src/java/lib/\*.jar:/usr/local/zookeeper-3.4.6/bin/../conf:.:/usr/java/jdk1.7.0\_80/lib/dt.jar:/usr/java/jdk1.7.0\_80/lib/tools.jar

2016-03-08 14:21:31,507 [myid:] - INFO [main:Environment@100] - Client environment:java.library.path=/usr/java/packages/lib/amd64:/usr/lib64:/lib64:/lib:/usr/lib

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:java.io.tmpdir=/tmp

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:java.compiler=<NA>

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:os.name=Linux

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:os.arch=amd64

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:os.version=3.10.0-123.9.3.el7.x86\_64

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:user.name=root

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:user.home=/root

2016-03-08 14:21:31,508 [myid:] - INFO [main:Environment@100] - Client environment:user.dir=/usr/local/zookeeper-3.4.6/bin

2016-03-08 14:21:31,510 [myid:] - INFO [main:ZooKeeper@438] - Initiating client connection, connectString=log1:2181 sessionTimeout=30000 watcher=org.apache.zookeeper.ZooKeeperMain$MyWatcher@ee01430

Welcome to ZooKeeper!

2016-03-08 14:21:31,534 [myid:] - INFO [main-SendThread(log1:2181):ClientCnxn$SendThread@975] - Opening socket connection to server log1/114.55.29.86:2181. Will not attempt to authenticate using SASL (unknown error)

2016-03-08 14:21:31,539 [myid:] - INFO [main-SendThread(log1:2181):ClientCnxn$SendThread@852] - Socket connection established to log1/114.55.29.86:2181, initiating session

JLine support is enabled

[zk: log1:2181(CONNECTING) 0] 2016-03-08 14:21:31,572 [myid:] - INFO [main-SendThread(log1:2181):ClientCnxn$SendThread@1235] - Session establishment complete on server log1/114.55.29.86:2181, sessionid = 0x25354db0d430000, negotiated timeout = 30000

WATCHER::

WatchedEvent state:SyncConnected type:None path:null

[zk: log1:2181(CONNECTED) 0]

## Window可视化客户端

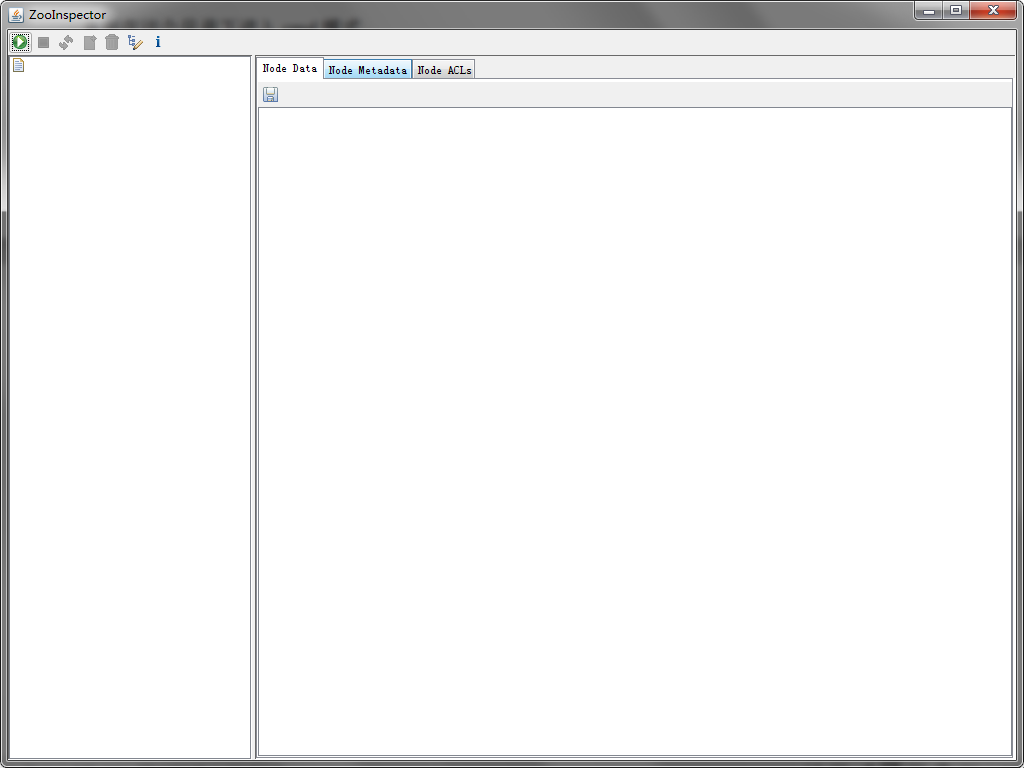
下载了windows可视化客户端

打开可以在里面的build里面发现一个jar包

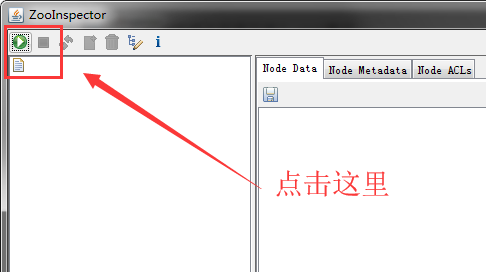


直接在这个目录下进入cmd模式

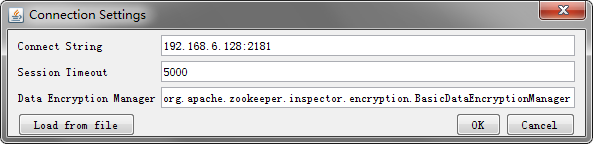
然后输入 java -jar zookeeper-dev-ZooInspector.jar 回车键开启客户端



点击



输入ip地址然后点击OK



最坑的地方来了，这里要等待一会儿，千万不要以为失败了，只要等着就好了

这里就会出现