四台主机上：/etc/profile中添加两个环境变量并使之生效：

export HDFS\_JOURNALNODE\_USER=root

export HDFS\_ZKFC\_USER=root

hdfs-site.xml

|  |
| --- |
| <property>  <name>dfs.nameservices</name>  <value>mycluster</value>  </property>  <property>  <name>dfs.ha.namenodes.mycluster</name>  <value>nn1,nn2,nn3</value>  </property>  <property>  <name>dfs.namenode.rpc-address.mycluster.nn1</name>  <value>node1:8020</value>  </property>  <property>  <name>dfs.namenode.rpc-address.mycluster.nn2</name>  <value>node2:8020</value>  </property>  <property>  <name>dfs.namenode.rpc-address.mycluster.nn3</name>  <value>node3:8020</value>  </property>  <property>  <name>dfs.namenode.shared.edits.dir</name>  <value>qjournal://node1:8485;node2:8485;node3:8485/mycluster</value>  </property>  <property>  <name>dfs.client.failover.proxy.provider.mycluster</name>  <value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider</value>  </property>  <property>  <name>dfs.ha.fencing.methods</name>  <value>sshfence</value>  </property>  <property>  <name>dfs.ha.fencing.ssh.private-key-files</name>  <value>/root/.ssh/id\_dsa</value>  </property>  <property>  <name>dfs.ha.automatic-failover.enabled</name>  <value>true</value>  </property> |

core-site.xml

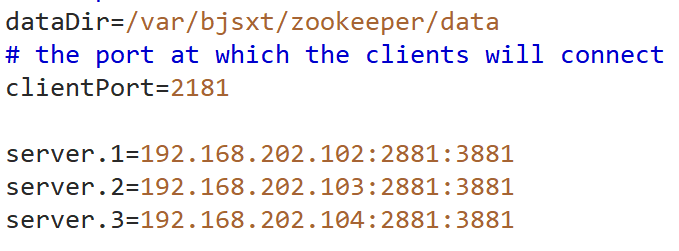
|  |
| --- |
| <property>  <name>fs.defaultFS</name>  <value>hdfs://mycluster</value>  </property>  <property>  <name>dfs.journalnode.edits.dir</name>  <value>/var/hadoop/ha/jnnedits</value>  </property>  <property>  <name>hadoop.tmp.dir</name>  <value>/var/hadoop/ha</value>  </property>  <property>  <name>ha.zookeeper.quorum</name>  <value>node2:2181,node3:2181,node4:2181</value>  </property> |

node2/node3/node4上传zookeeper-3.4.9.tar.gz

|  |
| --- |
| tar -zxf zookeeper-3.4.9.tar.gz -C /opt  cd /opt/zookeeper-3.4.9/conf |

复制zoo\_sample.cfg为zoo.cfg

编辑zoo.cfg



node2上执行

|  |
| --- |
| mkdir -p /var/bjsxt/zookeeper/data  echo 1 > /var/bjsxt/zookeeper/data/myid |

node3上执行

|  |
| --- |
| mkdir -p /var/bjsxt/zookeeper/data  echo 2 > /var/bjsxt/zookeeper/data/myid |

node4上执行

|  |
| --- |
| mkdir -p /var/bjsxt/zookeeper/data  echo 3 > /var/bjsxt/zookeeper/data/myid |

在node2/node3/node4上的/etc/profile中添加

|  |
| --- |
| ZOOKEEPER\_PREFIX=/opt/zookeeper-3.4.9  export PATH=$PATH:$ZOOKEEPER\_PREFIX/bin |

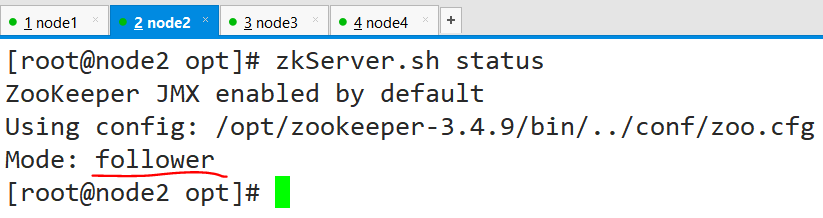
使之生效

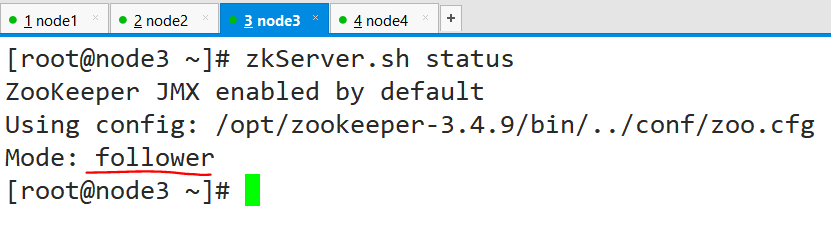
source /etc/profile

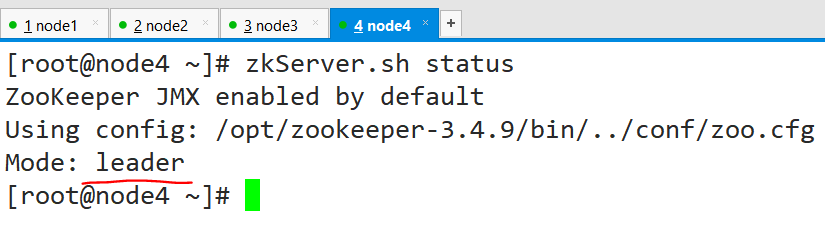
启动zk1/zk2/zk3

|  |
| --- |
| zkServer.sh start  zkServer.sh status  zkServer.sh stop |

结果：







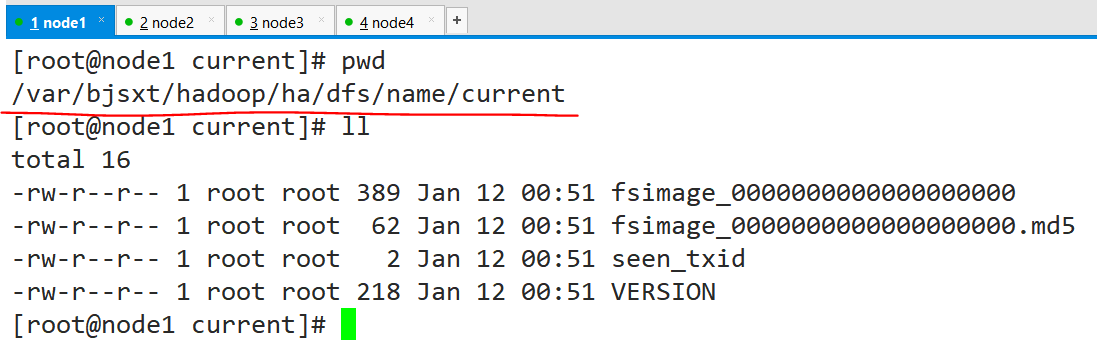
在

node1和node2和node3上执行：

hdfs --daemon start journalnode

启动JournalNodes

hdfs namenode -format

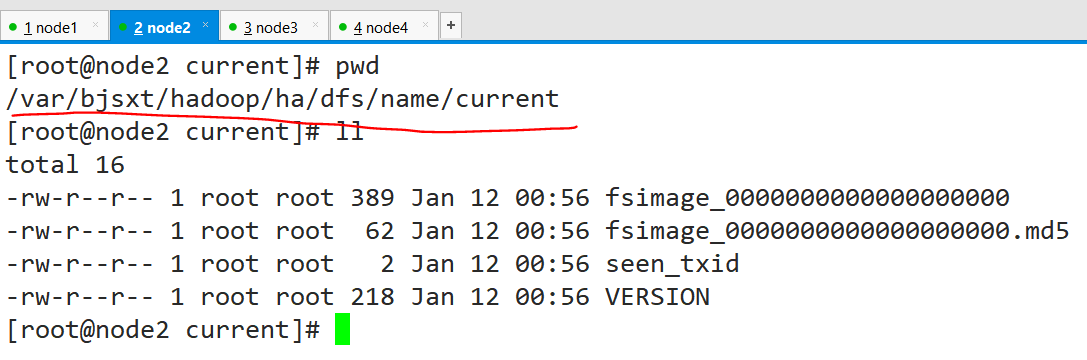


在node1上执行：

hdfs --daemon start namenode

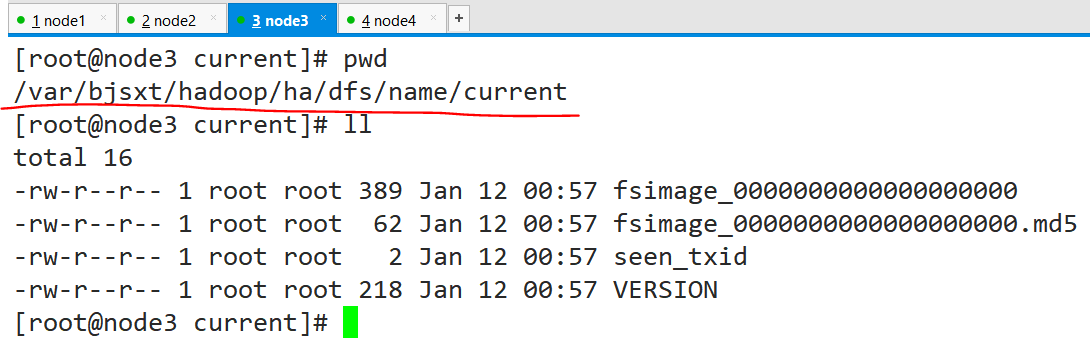
在node2上执行：

hdfs namenode -bootstrapStandby



在node3上执行：

hdfs namenode -bootstrapStandby

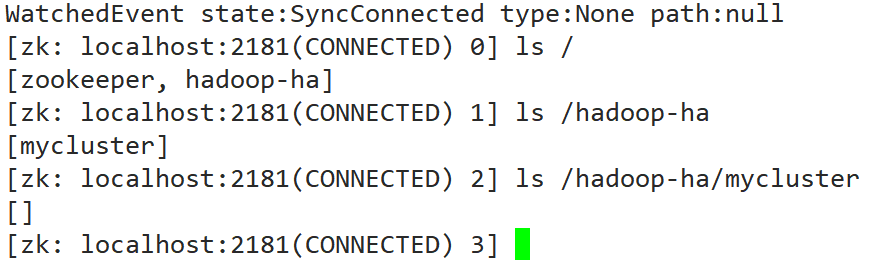


初始化zookeeper上NameNode的状态

在node1上执行：

hdfs zkfc -formatZK

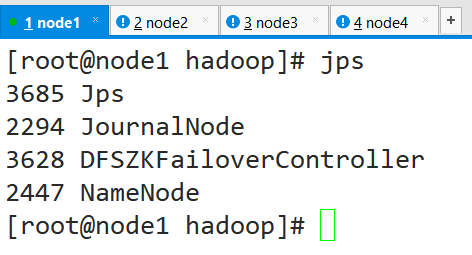
在node2或node3或node4上执行zkCli.sh：



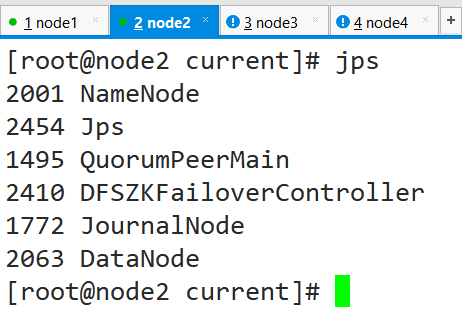
在node1或node2或node3上执行start-dfs.sh启动集群（记着先开启zookeeper集群）

执行jps查看进程信息：

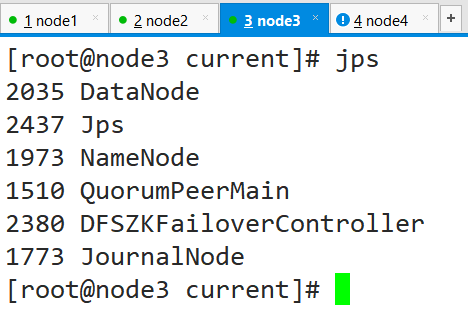
node1：



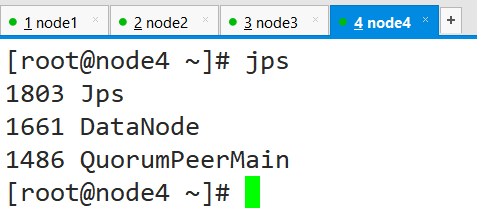
node2上：



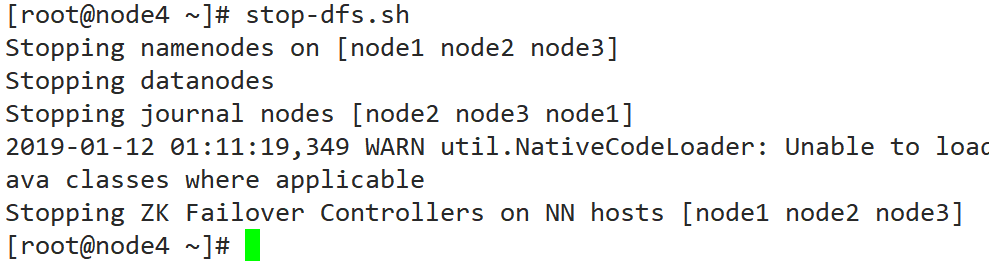
node3上：



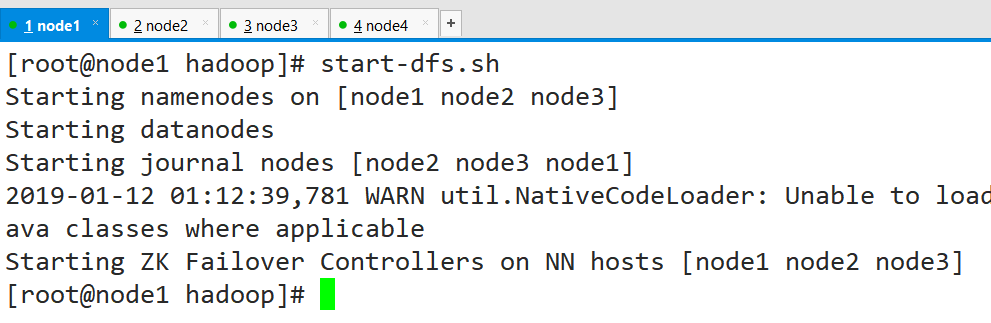
node4上：



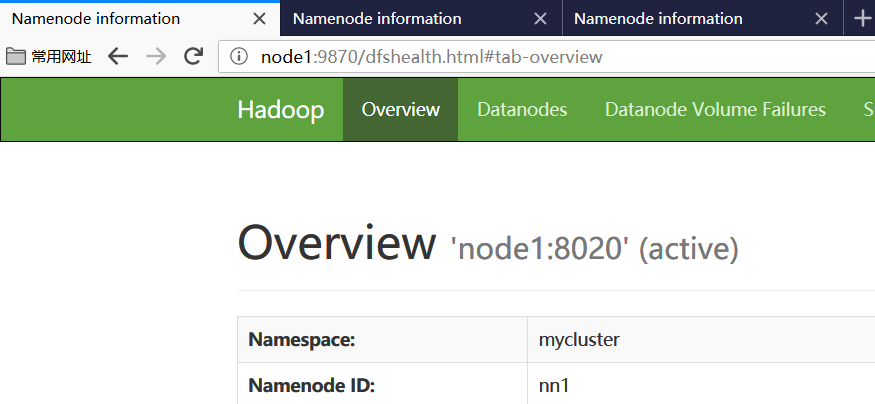
执行：stop-dfs.sh停止集群



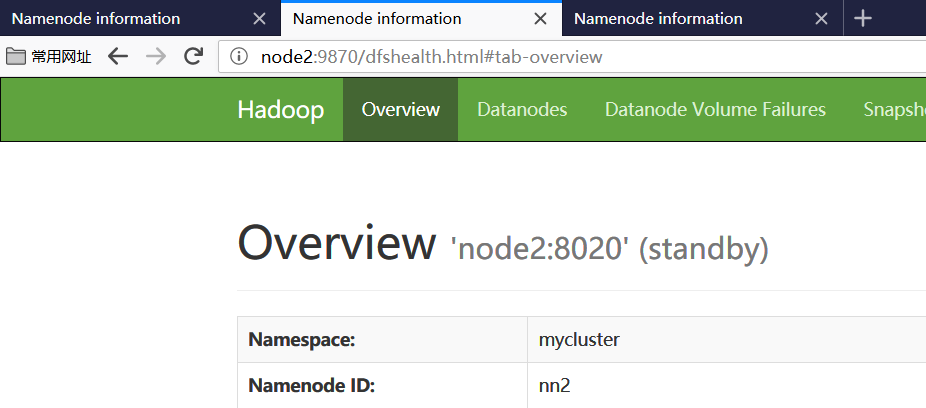
start-dfs.sh:



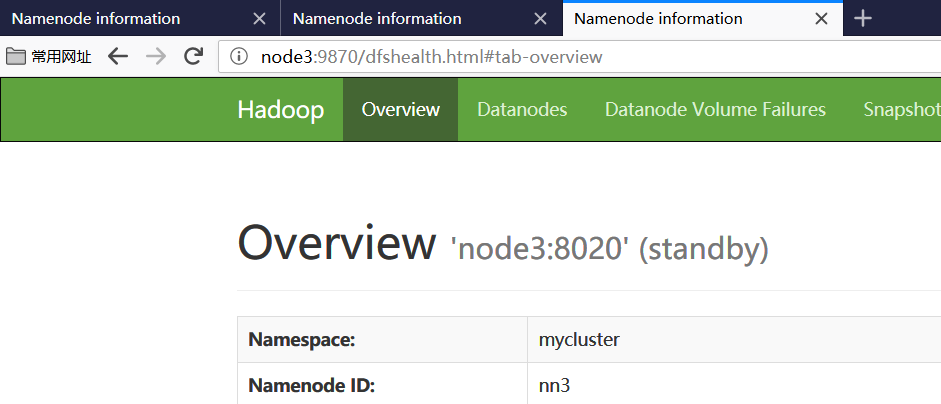
<http://node1:9870>



<http://node2:9870>



<http://node3:9870>



在node2或node3或node4执行zkCli.sh：

