zookeeper+hadoop+spark环境搭建与配置

整理：成都-项管-Jerry 2298456

目录

[2 zookeeper四台机器集群规划方案 1](#_Toc410770771)

[3 配置文件 1](#_Toc410770772)

[[root@sparkmaster hadoop]# vi core-site.xml 1](#_Toc410770773)

[4 启动Hadoop 6](#_Toc410770774)

[4.1 启动方法一 6](#_Toc410770775)

[5 Spark集群 7](#_Toc410770776)

[5.1 Spark环境配置 7](#_Toc410770777)

[5.2 启动Spark 10](#_Toc410770778)

[5.3 测试Spark 11](#_Toc410770779)

# zookeeper四台机器集群规划方案

集群规划：

|  |  |  |  |
| --- | --- | --- | --- |
| 主机名 | IP | 安装的软件 | 运行的进程 |
| sparkmaster | 192.168.1.10 | jdk、hadoop | ResourceManager、NameNode、DFSZKFailoverController(zkfc) |
| sparkworkerone | 192.168.1.11 | jdk、hadoop、zookeeper | ResourceManager、JournalNode、QuorumPeerMain、NameNode、DFSZKFailoverController(zkfc) |
| sparkworkertwo | 192.168.1.12 | jdk、hadoop、zookeeper | DataNode、NodeManager、JournalNode、QuorumPeerMain |
| sparkworkerthree | 192.168.1.13 | jdk、hadoop、zookeeper | DataNode、NodeManager、JournalNode、QuorumPeerMain |

# 配置文件

目录地址： /usr/src/bigdatasoft/hadoop-2.4.1/etc/hadoop/

1. 修改文件core-site.xml

[root@sparkmaster hadoop]# vi core-site.xml

|  |
| --- |
| <configuration>  <property>  <name>fs.defaultFS</name>  <value>hdfs://ns1</value>  </property>  <property>  <name>hadoop.tmp.dir</name>  <value>/usr/src/bigdatasoft/hadoop-2.4.1/tmp</value>  </property>  <property>  <name>ha.zookeeper.quorum</name>  <value>sparkworkerone:2181,sparkworkertwo:2181,sparkworkerthree:2181</value>  </property>  </configuration> |

1. 修改文件hdfs-site.xml

[root@sparkmaster hadoop]# vi hdfs-site.xml

|  |
| --- |
| <configuration>  <property>  <name>dfs.replication</name>  <value>2</value>  </property>  <property>  <name>dfs.permissions</name>  <value>false</value>  </property>  <property>  <name>dfs.permissions.enabled</name>  <value>false</value>  </property>  <property>  <name>dfs.nameservices</name>  <value>ns1</value>  </property>  <property>  <name>dfs.ha.namenodes.ns1</name>  <value>nn1,nn2</value>  </property>  <property>  <name>dfs.namenode.rpc-address.ns1.nn1</name>  <value>sparkmaster:9000</value>  </property>  <property>  <name>dfs.namenode.http-address.ns1.nn1</name>  <value>sparkmaster:50070</value>  </property>  <property>  <name>dfs.namenode.rpc-address.ns1.nn2</name>  <value>sparkworkerone: 9000</value>  </property>  <property>  <name>dfs.namenode.http-address.ns1.nn2</name>  <value>sparkworkerone:50070</value>  </property>  <property>  <name>dfs.namenode.servicerpc-address. ns1.nn1</name>  <value>sparkmaster:53310</value>  </property>  <property>  <name>dfs.namenode.servicerpc-address. ns1.nn2</name>  <value>sparkworkerone:53310</value>  </property>  <property>  <name>dfs.namenode.shared.edits.dir</name>  <value>qjournal:sparkworkerone:8485;sparkworkertwo:8485;sparkworkerthree:8485/ns1</value>  </property>  <property>  <name>dfs.journalnode.edits.dir</name>  <value>/usr/src/bigdatasoft/hadoop-2.4.1/journal</value>  </property>  <property>  <name>dfs.ha.automatic-failover.enabled</name>  <value>true</value>  </property>  <property>  <name>dfs.client.failover.proxy.provider.ns1</name>  <value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider</value>  </property>  <property>  <name>dfs.ha.fencing.methods</name>  <value>  sshfence  shell(/bin/true)  </value>  </property>  <property>  <name>dfs.ha.fencing.ssh.private-key-files</name>  <value>/root/.ssh/id\_rsa</value>  </property>  <property>  <name>dfs.ha.fencing.ssh.connect-timeout</name>  <value>30000</value>  </property>  <property>  <name>dfs.namenode.handler.count</name>  <value>100</value>  </property>  </configuration> |

1. 修改文件mapred-site.xml

[root@sparkmaster hadoop]#vi mapred-site.xml

|  |
| --- |
| <configuration>  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  </configuration> |

1. 修改文件yarn-site.xml

[root@sparkmaster hadoop]# vi yarn-site.xml

|  |
| --- |
| <configuration>  <property>  <name>yarn.resourcemanager.ha.enabled</name>  <value>true</value>  </property>  <property>  <name>yarn.resourcemanager.cluster-id</name>  <value>yarn-ha</value>  </property>  <property>  <name>yarn.resourcemanager.ha.rm-ids</name>  <value>rm1,rm2</value>  </property>  <property>  <name>yarn.resourcemanager.hostname.rm1</name>  <value>sparkmaster</value>  </property>  <property>  <name>yarn.resourcemanager.hostname.rm2</name>  <value>sparkworkerone</value>  </property>  <property>  <name>yarn.resourcemanager.zk-address</name>  <value>sparkworkerone:2181,sparkworkertwo:2181,sparkworkerthree:2181</value>  </property>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  </configuration> |

1. 修改文件slaves

[root@sparkmaster hadoop]# vi slaves

|  |
| --- |
| sparkworkertwo  sparkworkerthree |

# 启动Hadoop

## 启动方法一

以下命令都均是分别在 /usr/src/bigdatasoft/hadoop-2.4.1 目录下执行

1. 格式化zk集群

在sparkmaster/sparkworkerone上执行 bin/hdfs zkfc –formatZK

出现以下问题：



Does not contain a valid host:port authority: sparkworkerone :9000

Sparkworkerone 和 ：之间存在一个空格，去掉就解决了

1. 启动journalnode（分别在sparkworkerone、sparkworktwo、sparkworkerthree上执行）

sbin/hadoop-daemon.sh start journalnode

1. 格式化HDFS

格式化namenode、启动namenode

在sparkmaster/sparkworkerone上执行 bin/hdfs namenode -format

(需要再次重新执行时，可以先删除/usr/src/bigdatasoft/hadoop-2.4.1/ tmp/dfs/name下的文件)

在sparkmaster上分别执行 sbin/hadoop-daemon.sh start namenode

在sparkworkerone上执行 bin/hdfs namenode -bootstrapStandby

在sparkworkerone上分别执行 sbin/hadoop-daemon.sh start namenode

1. 启动datanode

在sparkworkertwo、sparkworkthree上分别执行sbin/hadoop-daemons.sh start datanode

1. 启动ZKFC（专门用来实现自动切换的，zookeeper 。。。控制器）哪里有namenode，在哪里就要启动ZKFC

在sparkmaster、sparkworkerone上 启动zkfc，执行命令 sbin/hadoop-daemon.sh start zkfc

1. 启动resourcemanager和nodemanager

在sparkmaster/sparkworkerone上执行 sbin/start-yarn.sh start resourcemanager; 或 sbin/yarn-daemon.sh start resourcemanager

# Spark集群

## Spark环境配置

1. 拷贝软件及解压

[root@sparkmaster bigdatasoft]#scp -r /usr/src/bigdatasoft/ 192.168.1.25:/usr/local/src

[root@sparkmaster bigdatasoft]# scp spark-1.2.0-bin-hadoop2.4.tgz 192.168.1.25:/usr/local/src

[root@sparkmaster bigdatasoft]# scp scala-2.10.4.tgz 192.168.1.25:/usr/local/src

[root@sparkmaster bigdatasoft]# scp scala-SDK-3.0.3-2.10-linux.gtk.x86\_64.tar.gz 192.168.1.25:/usr/local/src

cd /usr/src/bigdatasoft/

tar -zxvf scala-SDK-3.0.3-2.10-linux.gtk.x86\_64.tar.gz

tar -zxvf scala-2.10.4.tgz

tar -zxvf spark-1.2.0-bin-hadoop2.4.tgz

scp spark-1.2.0-bin-hadoop2.4.tgz 192.168.1.11:/usr/src/bigdatasoft/

scp /etc/profile 192.168.1.12:/etc/

1. 增加scala和spark的环境变量参数：

[root@sparkmaster ~]# vi /etc/profile

export JAVA\_HOME=/usr/src/bigdatasoft/jdk1.7.0\_67

export JRE\_HOME=/usr/src/bigdatasoft/jdk1.7.0\_67/jre

export HADOOP\_HOME=/usr/src/bigdatasoft/hadoop-2.4.1

export SCALA\_HOME=/usr/src/bigdatasoft/scala-2.10.4

export SPARK\_HOME=/usr/src/bigdatasoft/spark-1.2.0-bin-hadoop2.4

export PATH=.:$JAVA\_HOME/bin:$JAVA\_HOME/jre/bin:$HADOOP\_HOME/bin:$SCALA\_HOME/bin:$SPARK\_HOME/bin:$PATH

export CLASSPATH=$CLASSPATH:.:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib:$HADOOP\_HOME/lib:$SCALA\_HOME/lib:$SPARK\_HOME/lib

source /etc/profile

查看scala 版本信息：scala  -version

1. 配置Spark环境
2. 配置spark-env.sh文件

把spark-env.sh.template 拷贝到spark-env.sh：cp spark-env.sh.template spark-env.sh

[root@sparkmaster conf]# vim spark-env.sh

添加以下内容：

export HADOOP\_CONF\_DIR=/usr/src/bigdatasoft/hadoop-2.4.1/etc/hadoop

export JAVA\_HOME=/usr/src/bigdatasoft/jdk1.7.0\_67

export SCALA\_HOME=/usr/src/bigdatasoft/scala-2.10.4

export SPARK\_MASTER\_IP=192.168.1.10

export SPARK\_WORKER\_MEMORY=1g

1. 配置slaves文件

把slaves.template拷贝到slaves: cp slaves.template slaves

添加以下内容：

sparkworkerone

sparkworkertwo

sparkworkerthree

1. 将spark-env.sh 和slaves文件拷贝到三台worker机器

查看路径 [root@sparkmaster conf]# pwd

/usr/src/bigdatasoft/spark-1.2.0-bin-hadoop2.4/conf

拷贝文件

scp spark-env.sh 192.168.1.11:/usr/src/bigdatasoft/spark-1.2.0-bin-hadoop2.4/conf

scp slaves 192.168.1.13:/usr/src/bigdatasoft/spark-1.2.0-bin-hadoop2.4/conf

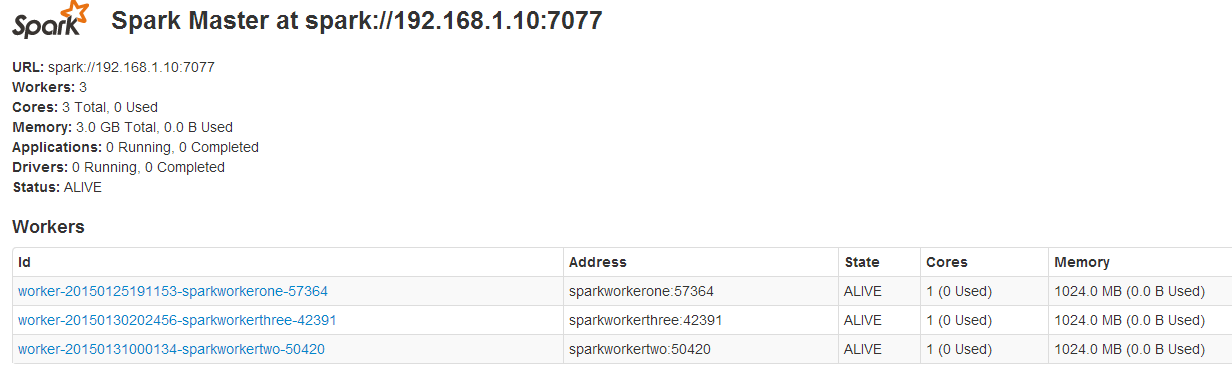
## 启动Spark

（路径：/usr/src/bigdatasoft/spark-1.2.0-bin-hadoop2.4/sbin）

[root@sparkmaster spark-1.2.0-bin-hadoop2.4]# sbin/start-all.sh

访问地址：http://192.168.1.10:8080/ 或 http://sparkmaster:8080/

Spark集群的web页面信息



查看其他相关环境参数

<http://192.168.1.10:4041/>

<http://192.168.1.10:4041/environment/>

<http://192.168.1.10:4041/executors/>

http://192.168.1.10:4041/stages/

进入spark shell控制台

[root@sparkmaster bin]# spark-shell

## 测试Spark

把spark 安装目录的”README.md” 复制到HDFS系统里

[root@sparkmaster spark-1.2.0-bin-hadoop2.4]# hadoop dfs -copyFromLocal README.md /home

或

[root@sparkmaster spark-1.2.0-bin-hadoop2.4]# hadoop fs -put README.md / /usr

查看上传后的文件

[root@sparkmaster spark-1.2.0-bin-hadoop2.4]# hadoop fs -ls /

读取README.md 文件内容

scala> val file = sc.textFile("hdfs://sparkmaster:9000/home/README.md")

scala> val sparks = file.filter(line => line.contains("Spark"))

scala> sparks.count //统计”spark”单词出现的次数

验证统计次数的准确性

[root@sparkmaster spark-1.2.0-bin-hadoop2.4]# grep Spark README.md|wc

19 158 1248

使用spark的 cache

scala> sparks.cache