# 配置Spark集群的点点滴滴

## 集群拓扑环境

|  |  |  |
| --- | --- | --- |
| Hostname | IP | Role |
| buaa | 192.168.108.211 | NameNode/DataNode/SecondaryNode  ResourceManager/NodeManger  Master/Worker |

## 设置静态IP和主机名

### 设置静态IP

以CentOS7为例。

|  |
| --- |
| 1. 编辑IP的配置文件   vim /etc/sysconfig/network-scripts/ifcfg-enp0s3   1. 将配置文件更改如下：   BOOTPROTO=”static” #将IP改为静态配置  IPADDR=”IP地址” #查看windows下连接的网段  GATEWAY=”网关”  NETMASK=”子网掩码”  DNS1=”网关”  DNS2=”180.76.76.76” #百度的DNS解析服务器   1. 重启网络服务   service network restart   1. 查看网络配置   ip addr |

### 配置主机名

|  |
| --- |
| 1. 编辑本主机名文件   vim /etc/hostname   1. 将文件中的主机名更改为：   buaa   1. 编辑主机名文件   vim /etc/hosts   1. 将文件中的内容更改为：   IP buaa #这里的IP是2.1中配置的静态IP |

## 配置SSh免登陆

|  |
| --- |
| 1. 生成密钥对   ssh-keygen –t rsa   1. 默认密钥存储文件   回车（Enter）   1. 输入SSH登录密码   回车（Enter） #免登陆，不输入密码  回车（Enter） #确定密码   1. 将公钥保存到认证文件中   cat ~/.ssh/id\_rsa.pub >> authorized\_keys   1. 设置.ssh目录权限   chmod 700 ~/.ssh   1. 设置私钥文件的权限   chmod 600 ~/.ssh/id\_rsa   1. 设置其余文件权限   chmod 644 ~/.ssh/id\_rsa.pub  chmod 644 ~/.ssh/authorized\_keys |

## 安装所需软件

### Java配置

#### 4.1.1删除openJDK

|  |
| --- |
| 1. 查询已安装的openJDK  rpm –qa | grep jdk  2. 删除已安装的openJDK  rpm –e –nodeps xxxx #这里的“xxx”是上一条命令查询到的结果 |

#### 下载JDK

|  |
| --- |
| 1. 下载JDK免安装版的压缩包：   <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>   1. 将压缩包(jdk-8u77-linux-x64.tar.gz)解压：   tar –zxf jdk-8u77-linux-x64.tar.gz #解压可以得到jdk1.8.0\_77文件目录   1. 将jdk1.8.0\_77目录移至用户软件安装空间（将所有自己的软件放在一个文件夹下）   mv jdk1.8.0\_77 otherDir |

### 下载Scala

|  |
| --- |
| 1. 下载Scala免安装版的压缩包 2. http://www.scala-lang.org/download/2.11.8.html 3. 将压缩包（scala-2.11.8.tgz）解压：   tar –zxf scala-2.11.8.tgz #解压可以得到scala-2.11.8文件目录   1. 将scala-2.11.8目录移至用户软件安装空间   mv scala-2.11.8 otherDir |

### 下载Hadoop

|  |
| --- |
| 1. 选择Hadoop版本下载：这里使用2.5.2版本   <http://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.5.2/hadoop-2.5.2.tar.gz>   1. 将压缩包（hadoop-2.5.2.tar.gz）解压：   tar –zxf hadoop-2.5.2.tar.gz #解压可以得到hadoop-2.5.2文件目录   1. 将hadoop-2.5.2目录移至用户软件安装空间 |

### 下载Spark

|  |
| --- |
| 1. 选择Spark版本下载：这里使用1.2.1版本   <http://archive.apache.org/dist/spark/spark-1.2.1/spark-1.2.1-bin-hadoop2.4.tgz>   1. 将压缩包（spark-1.2.1-bin-hadoop2.4.tgz）解压：   tar –zxf spark-1.2.1-bin-hadoop2.4 #解压可以得到spark-1.2.1-bin-hadoop2.4文件目录   1. 将spark-1.2.1-bin-hadoop2.4目录移至用户软件安装空间 |

## 配置软件

### 软件路径配置

|  |
| --- |
| 1. 编辑.bashrc文件   vim ~/.bashrc   1. 向.bashrc文件尾添加一下信息   export JAVA\_HOME=JDK的安装目录  export CLASSPATH=.:${JAVA\_HOME}/jre/rt.jar:${JAVA\_HOME}/lib/dt.jar:${JAVA\_HOME}/lib/tools.jar  export SCALA\_HOME=Scla的安装目录  export HADOOP2\_HOME=Hadoop的安装目录  export SPARK1\_HOME=Spark的安装目录  export PATH=${PATH}:${JAVA\_HOME}/bin:${SCALA\_HOME}/bin:${HADOOP2\_HOME}/bin:${HADOOP2\_HOME}/sbin:${SPARK1\_HOME}/bin:${SPARK1\_HOME}/sbin   1. 重新读取并执行.bashrc中的命令   source ~/.bashrc   1. 检测各软件路径是否配置正确   java –version  scala –version  hadoop version |

### 配置并操作Hadoop分布式集群

#### 创建hdfs的数据存储目录

|  |
| --- |
| 1. 创建hdfs的存储目录   mkdir /home/…/Hadoop的安装目录/storage   1. 创建存放名称数据的目录   mkdir /home/…/Hadoop的安装目录/storage/name   1. 创建存放应用数据的目录 2. mkdir /home/…/Hadoop的安装目录/storage/data |

#### 配置core-site.xml

注：当前工作目录pwd为/…/Hadoop安装目录/etc/hadoop

1. 编辑core-size.xml

vim ./core-site.xml

1. 向core-site.xml文件添加以下内容

|  |
| --- |
| <configuration>  <!-- 设置HDFS的节点名称：就是告诉集群，HDFS节点在哪 -->  <property>  <name>fs.defaultFS</name>  <value>hdfs://buaa:8020</value>  <final>true</final>  </property>  <!-- 设置临时文件夹 -->  <property>  <name>hadoop.tmp.dir</name>  <value>/home/yangel/Softwares/hadoop-2.5.2/tmp/hadoop-${user.name}</value>  </property>  <property>  </property> |

#### 配置hdfs-site.xml

注：当前工作目录pwd为/…/Hadoop安装目录/etc/hadoop

1. 编辑hdfs-size.xml

vim ./hdfs-site.xml

1. 向hdfs-site.xml文件添加以下内容

|  |
| --- |
| <configuration>  <!—设置第二名称节点的地址：<主机名/IP：端口> -->  <property>  <name>hdfs.namenode.secondary.http-address</name>  <value>buaa:50090</value>  </property>  <!—设置名称节点中元信息（名称数据）的存储位置 -->  <property>  <name>dfs.namenode.name.dir</name>  <value>/home/yangel/Softwares/hadoop-2.5.2/storage/name</value>  </property>  <!-- 设置数据节点中数据的存储位置 -->  <property>  <name>dfs.datanode.data.dir</name>  <value>/home/yangel/Softwares/hadoop-2.5.2/storage/data</value>  </property>  <!-- 设置是否启动Web访问HDFS-->  <property>  <name>dfs.webhdfs.enabled</name>  <value>true</value>  </property>  <!-- 设置是否启动dfs访问权限控制 -->  <property>  <name>dfs.permission</name>  <value>false</value>  </property>  <!-- 设置数据副本数 -->  <property>  <name>dfs.replication</name>  <value>1</value>  </property>  </configuration> |

#### 配置mapred-site.xml

注：当前工作目录pwd为/…/Hadoop安装目录/etc/hadoop

1. 从mapred-site.xml.template创建mapred-size.xml

cp ./mapred-site.xml.template ./mapred-site.xml

1. 编辑mapred-site.xml

vim mapred-site.xml

1. 向mapred-site.xml文件添加以下内容

|  |
| --- |
| <configuration>  <!—设置资源调度框架-->  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  <!—设置历史作业：就是访问历史作业数据的位置：<主机名/IP:端口号> -->  <property>  <name>mapreduce.jobhistory.address</name>  <value>buaa:10020</value>  </property>  <!-- 设置历史作业的webapp地址 -->  <property>  <name>mapreduce.jobhistory.webapp.address</name>  <value>buaa:10021</value>  </property>  <property>  <name>mapreduce.jobhistory.intermediate-done-dir</name>  <value>/home/yangel/Softwares/hadoop-2.5.2/tmp</value>  </property>  <property>  <name>mapreduce.jobhistory.done-dir</name>  <value>/home/yangel/Softwares/hadoop-2.5.2/tmp</value>  </property>  </configuration> |

#### 配置yarn-site.xml

注：当前工作目录pwd为/…/Hadoop安装目录/etc/hadoop

1. 编辑yarn-size.xml

vim ./yarn-site.xml

1. 向yanr-site.xml文件添加以下内容

|  |
| --- |
| <configuration>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <property>  <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>  <value>org.apache.hadoop.mapred.ShuffleHandler</value>  </property>  <!—设置资源管理器的地址-->  <property>  <name>yarn.resourcemanager.address</name>  <value>buaa:8030</value>  </property>  <!—设置调度器的地址->  <property>  <name>yarn.resourcemanager.scheduler.address</name>  <value>buaa:8031</value>  </property>  <!—设置资源跟踪器的地址-->  <property>  <name>yarn.resourcemanager.resource-tracker.address</name>  <value>buaa:8032</value>  </property>  <!—设置资源管理器的管理员地址-->  <property>  <name>yarn.resourcemanager.admin.address</name>  <value>buaa:8033</value>  </property>  <!—设置资源管理器的web访问地址-->  <property>  <name>yarn.resourcemanager.webapp.address</name>  <value>buaa:8034</value>  </property>  </configuration> |

#### 配置主节点文件masters

注：当前工作目录pwd为/…/Hadoop安装目录/etc/hadoop

1. 创建masters文件

touch masters

1. 编辑masters文件添加以下内容

vim masters

|  |
| --- |
| buaa # 当前节点是主节点 |

#### 配置从节点文件

注：当前工作目录pwd为/…/Hadoop安装目录/etc/hadoop

1. 编辑slaves文件并添加以下内容

|  |
| --- |
| buaa # 当前节点也是从节点 |

#### 格式化HDFS

|  |
| --- |
| 1. 执行以下命令格式化HDFS   hdfs namenode -format |

#### 启动Hadoop集群

|  |
| --- |
| 1. 启动HDFS   start-dfs.sh   1. 检测HDFS是否启动成功   jps #执行jps命令，如果出现NameNode/DataNode/SecondaryNamNode即为成功   1. 启动YARN   start-yarn.sh   1. 检测YARN是否启动成功   jps #执行jps命令，如果出现ResourceManager,NodeManager即为成功 |

#### 操作Hadoop集群

|  |
| --- |
| 1. 查看Hadoop分布式文件系的概要   <http://buaa:50070/dfshealth.html#tab-overview>   1. 查看Hadoop的历史作业记录   <http://buaa:8034/cluster> |

#### 关闭Hadoop集群

|  |
| --- |
| 1. 关闭HDFS   stop-dfs.sh   1. 关闭YARN   stop-yarn.sh |

### 配置并操作Spark集群

#### 配置slaves文件

注：当前工作目录pwd为/…/Spark安装目录/conf

1. 从slaves.template创建slaves文件

cp ./slaves.template slaves

1. 编辑slaves文件

vim slaves

1. 向slaves文件添加以下内容

|  |
| --- |
| buaa #Spark的worker节点 |

#### 配置spark-env.sh文件

注：当前工作目录pwd为/…/Spark安装目录/conf

1. 从spark-env.template创建spark-env文件

cp ./ spark-env.template spark-env

1. 编辑spark-env文件

vim spark-env

1. 向spark-env文件添加以下内容

|  |
| --- |
| export JAVA\_HOME=JDK的安装目录  export SCALA\_HOME=Scala的安装目录  export SPARK\_MASTER\_IP=主节点的IP地址  export SPARK\_WORKER\_MEMORY=1g #工作节点的内存大小  export HADOOP\_CONF\_DIR=/../Hadoop安装目录/etc/hadoop #hadoop的配置文件目录 |

#### 启动Spark集群

|  |
| --- |
| 1. 启动主节点，在主节点上执行以下命令   start-master.sh   1. 启动从节点，在从节点上执行以下命令   start-slave.sh |

#### 操作Spark集群

|  |
| --- |
| 1. 查看Spark主节点的信息   http://192.168.108.211:8080/ |

#### 关闭Spark集群

|  |
| --- |
| 1. 关闭从节点，在从节点上执行以下命令   stop-slave.sh   1. 关闭主节点，在主节点上执行以下命令   stop-master.sh |