# Hadoop安装手册

## 系统配置

### 修改主机名

在所有的主机上修改/etc/sysconfig/network文件中的HOSTNAME=hadoop1(hadoop2、hadoop3等)。

### 配置IP地址

修改/etc/sysconfig/network-scripts/ifcfg-eth0，修改后的信息如下：

|  |
| --- |
| DEVICE="eth0"  BOOTPROTO="static" #修改静态地址  HWADDR="00:0C:29:23:C0:B2"#MAC地址  IPV6INIT="no"#禁用IPV6  NM\_CONTROLLED="yes"  ONBOOT="yes"  TYPE="Ethernet"  IPADDR="192.168.204.130" #实际的IP地址，根据自己的网络情况进行设置  NETMASK="255.255.255.0" #子网掩码  GATEWAY="192.168.204.2" #网管  DNS1="8.8.8.8" #DNS  UUID="476b3e36-f94f-4d6e-8835-93da6035a00f" |

### 主机名与IP地址做映射

把所有节点的IP地址与主机名做映射后写入到/etc/hosts文件，然后分发到所有主机，样例如下：

|  |
| --- |
| 127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4  ::1 localhost localhost.localdomain localhost6 localhost6.localdomain6  192.168.204.130 hadoop1  192.168.204.131 hadoop2  192.168.204.132 hadoop3 |

### 关闭防火墙

在所有主机节点上关闭防火墙，命令如下：

|  |
| --- |
| chkconfig iptables off |

### 设置selinux

修改所有主机/etc/selinux/config文件SELINUX=disabled。

## 安装JDK

从网上下载jdk1.7以上64位linux版本。在此，我们安装jdk-7u67-linux-x64.rpm版

### 1、安装

通过rpm -ivh jdk-7u67-linux-x64.rpm命令进行安装；

### 2、验证

执行java –version命令，如果显示下边信息，说明安装成功。

|  |
| --- |
| java version "1.7.0\_67"  Java(TM) SE Runtime Environment (build 1.7.0\_67-b01)  Java HotSpot(TM) 64-Bit Server VM (build 24.65-b04, mixed mode) |

## 安装Hadoop

### 1、Hadoop的下载和解压

我从网上下载Hadoop的安装包，然后在所有的主机上解压即可（本次我们下载的hadoop-2.6.0-cdh5.10.1.tar.gz）。解压命令如下：

|  |
| --- |
| tar –xvf hadoop-2.6.0-cdh5.10.1.tar.gz |

### 2、Hadoop系统环境变量的配置

在所有主机的/etc/profile文件中添加下边的信息

|  |
| --- |
| export HADOOP\_HOME=/root/hadoop-2.6.0-cdh5.10.1  export PATH=$PATH:$HADOOP\_HOME/bin |

注释：/root/hadoop-2.6.0-cdh5.10.1实我们Hadoop的解压路径

## 配置Hadoop

在所有的主机上做以下配置（也可以在一台主机上配置完后，拷贝到其他主机）。所有配置的都在/root/hadoop-2.6.0-cdh5.10.1/etc/hadoop目录下（Hadoop解压后目录里的etc/hadoop里）。

### 1、core-site.xml的配置

|  |
| --- |
| <configuration>  <property>  <name>fs.defaultFS</name> #hdfs访问的唯一入口  <value>hdfs://hadoop1:8020</value>  </property>  </configuration> |

### 2、hdfs-site.xml文件的配置

|  |
| --- |
| <configuration>  <property>  <name>dfs.permissions.superusergroup</name>  <value>root</value>  </property>  <property>  <name>dfs.namenode.name.dir</name>  <value>/data/nn</value>  </property>  <property>  <name>dfs.datanode.data.dir</name>  <value>/data/dn</value>  </property>  <property>  <name>dfs.namenode.http-address</name>  <value>hadoop1:50070</value>  <description>  The address and the base port on which the dfs NameNode Web UI will listen.  </description>  </property>  <property>  <name>dfs.secondary.http-address</name>  <value>hadoop2:50070</value>  <description>  The address and the base port on which the dfs NameNode Web UI will listen.  </description>  </property>  </configuration> |

### 3、mapred-site.xml配置

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

### 4、yarn-site.xml配置

|  |
| --- |
| <configuration>  <!-- Site specific YARN configuration properties -->  <property>  <name>yarn.resourcemanager.hostname</name>  <value>hadoop3</value>  </property>  <property>  <description>Classpath for typical applications.</description>  <name>yarn.application.classpath</name>  <value>  $HADOOP\_CONF\_DIR,  $HADOOP\_COMMON\_HOME/\*,$HADOOP\_COMMON\_HOME/lib/\*,  $HADOOP\_HDFS\_HOME/\*,$HADOOP\_HDFS\_HOME/lib/\*,  $HADOOP\_MAPRED\_HOME/\*,$HADOOP\_MAPRED\_HOME/lib/\*,  $HADOOP\_YARN\_HOME/\*,$HADOOP\_YARN\_HOME/lib/\*  </value>  </property>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <property>  <name>yarn.nodemanager.local-dirs</name>  <value>/data/yarn/local</value>  </property>  <property>  <name>yarn.nodemanager.log-dirs</name>  <value>/data/yarn/logs</value>  </property>  <property>  <name>yarn.log.aggregation-enable</name>  <value>true</value>  </property>  <property>  <description>Where to aggregate logs</description>  <name>yarn.nodemanager.remote-app-log-dir</name>  <value>/var/log/hadoop-yarn/apps</value>  </property>  </configuration> |

### 添加JAVA\_HOME到hadoop-env.sh文件里

添加内容如下：

|  |
| --- |
| export JAVA\_HOME=/usr/java/default |

### 添加所有的主机名到slaves文件里，如下所示：

|  |
| --- |
| hadoop1  hadoop2  hadoop3 |

## 格式化namenode

在namenode所在主机（我们本次所规划的hadoop1）执行命令hadoop namenode –format就行格式化。

## 启动所进程

我们的启动方式有两种，一种是单独启动，也就是分别进入到每一台主机，一个一个进程启动。另外一种是批量启动，也就是在任一主机上通过两个脚本就可以把所有进程全部拉起来。

### 单独启动

|  |
| --- |
| hadoop1:  cd /root/hadoop-2.6.0-cdh5.10.1/sbin  ./hadoop-daemon.sh start namenode  ./hadoop-daemon.sh start datanode  ./yarn-daemon.sh start nodemanager  hadoop2:  cd /root/hadoop-2.6.0-cdh5.10.1/sbin  ./hadoop-daemon.sh start secondarynamenode  ./hadoop-daemon.sh start datanode  ./hadoop-daemon.sh start nodemanage  hadoop3:  cd /root/hadoop-2.6.0-cdh5.10.1/sbin  ./hadoop-daemon.sh start datanode  ./yarn-daemon.sh start resourcemanager  ./yarn-daemon.sh start nodemanager |

### 批量启动

在任一主机上（所在主机上的slaves文件必须配置）进入/root/hadoop-2.6.0-cdh5.10.1/sbin目录后执行./ start-dfs.sh和./start-yarn.sh。

## 验证集群是否成功

### 通过web界面来查看集群

在浏览器中输入<http://hadoop1:50070>和http://hadoop3:8088，能够看到如下界面，说明机器启动成功。

### 通过命令来验证

例如：通过hadoop fs –mkdir /test来创建测试路径，如果能创建就说明成功。

## 停止所有进程

在主机上（本主机的slaves文件进行过配置）执行./stop-dfs.sh来停止Hadoop相关进程，执行./ stop-yarn.sh来停止Yarn的相关进程（在执行命令时，要先进入到/root/hadoop-2.6.0-cdh5.10.1/sbin目录）。