

基于 CentOS 平台的 hadoop 集群配置与安装



选择进入 SSH 桌面

```
root@master:~  
login as: root  
root@202.197.41.232's password:  
Last login: Wed Mar 25 05:13:39 2015 from datanode1
```

用户名 root 密码 admin123 登陆

1 安装 jdk

下载并安装 jdk

```
yum install java-1.7.0-openjdk-devel
```

```
[root@master ~]# yum install java-1.7.0-openjdk-devel
```

验证成功安装

```
java -version
```

```
[root@master ~]# java -version  
java version "1.7.0_75"  
OpenJDK Runtime Environment (rhel-2.5.4.2.el7_0-x86_64 u75-b13)  
OpenJDK 64-Bit Server VM (build 24.75-b04, mixed mode)
```

2 安装 hadoop

下载 hadoop，并解压到/root/soft/apache/hadoop 目录下

```
cd ~
mkdir soft
cd soft/
mkdir apache
cd apache/
mkdir hadoop
cd hadoop/
wget http://mirrors.sonic.net/apache/hadoop/common/hadoop-2.6.0/hadoop-2.6.0.tar.gz
tar xvzf hadoop-2.6.0.tar.gz
```

2.2 配置环境变量

2.2.1 配置全局环境变量

修改 ~/.bashrc 文件

```
vi ~/.bashrc
```

在文件末尾加入下面配置信息：

```
export JAVA_HOME=/usr/lib/jvm/java
export HADOOP_HOME=/root/soft/apache/hadoop/hadoop-2.6.0
export HADOOP_CONFIG_HOME=$HADOOP_HOME/etc/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
```

如果不知道 java 配置位置可以使用以下命令查看

```
update-alternatives --config java
```

2.2.2 新建三个文件夹

temp：作为 Hadoop 的临时目录。

路径为：/root/soft/apache/hadoop/hadoop-2.6.0/tmp

Namenode：作为 NameNode 的存放目录。

路径为 /root/soft/apache/hadoop/hadoop-2.6.0/namenode

DataNode：作为 dataNode 的存放目录。

路径为：/root/soft/apache/hadoop/hadoop-2.6.0/datanode

```
cd $HADOOP_HOME/
mkdir tmp
cd tmp/
cd ../
mkdir namenode
```

```
cd namenode/  
cd ../  
mkdir datanode  
cd datanode/
```

修改 core-site.xml 配置

打开 core-site.xml 文件所在位置:

```
cd $HADOOP_HOME  
cd etc/hadoop
```

修改文件:

```
vi core-site.xml
```

输入 i 进入编辑:

在<configuration></configuration>之间添加

```
<configuration>  
  <property>  
    <name>hadoop.tmp.dir</name>  
    <value>/root/soft/apache/hadoop/hadoop-2.6.0/tmp</value>  
    <description>A base for other temporary directories.</description>  
  </property>  
  <property>  
    <name>fs.default.name</name>  
    <value>hdfs://master:9000</value>  
  </property>  
</configuration>
```

注意: hadoop.tmp.dir 值为/root/soft/apache/hadoop/hadoop-2.6.0/tmp
fs.default.name 配置为 hdfs://master:9000, 指向一个 Master 节点的主机。

2.2.4 修改 hdfs-site.xml 配置

```
vi hdfs-site.xml
```

输入 i 进入编辑:

```
<configuration>  
  <property>  
    <name>dfs.replication</name>  
    <value>2</value>  
  </property>  
  <property>  
    <name>dfs.namenode.name.dir</name>  
    <value>/root/soft/apache/hadoop/hadoop-2.6.0/namenode</value>  
  </property>  
  <property>  
    <name>dfs.datanode.data.dir</name>  
    <value>/root/soft/apache/hadoop/hadoop-2.6.0/datanode</value>  
  </property>  
</configuration>
```

2.2.5 修改 Mapred-site.xml 配置

(1) Hadoop 安装文件中提供了一个 mapred-site.xml.template 模板，将模板复制到 mapred-site.xml

```
cp mapred-site.xml.template mapred-site.xml
```

(2) 编辑 Mapred-site.xml 文件

```
vi mapred-site.xml
```

输入 i 进入编辑:

```
<configuration>
  <property>
    <name>mapred.job.tracker</name>
    <value>master:9001</value>
  </property>
</configuration>
```

2.2.6 修改 hadoop-env.sh 配置

```
vi hadoop-env.sh
```

输入 i 进入编辑

更改 JAVA_HOME 值

```
# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java
```

3 建立三个节点互访

3.1 更改/etc/hostname 文件

分别登陆 master, datanode1, datanode2 节点，输入:

```
vi /etc/hostname
```

将文件中的 localhost 分别改为: master, datanode1, datanode2

3.2 更改/etc/hosts 文件

(1) 分别登陆 master, datanode1, datanode2 节点，用 ifconfig 命令查看 ip 地址

```
[root@master hadoop]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    net 10.10.12.138 netmask 255.255.240.0 broadcast 10.10.15.255
    inet6 fe80::216:3eff:fe00:378 prefixlen 64 scopeid 0x20<link>
    ether 00:16:3e:00:03:78 txqueuelen 1000 (Ethernet)
    RX packets 7130052 bytes 483573553 (461.1 MiB)
    RX errors 0 dropped 1 overruns 0 frame 0
    TX packets 382379 bytes 2291399809 (2.1 GiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 21327 bytes 3627121 (3.4 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 21327 bytes 3627121 (3.4 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@master hadoop]# █
```

(2) 进入 master 节点，编辑 hosts 文件

```
vi /etc/hosts
```

在文件末尾添加

```
10.10.12.138 master
10.10.12.131 datanode1
10.10.12.130 datanode2
```

优点： 可以直接使用 datanode1 代替 ip 地址访问

(3) 将 hosts 及 bashrc 文件同步到 datanode1, datanode2 节点

```
scp /etc/hosts/ root@10.10.12.131:/etc/hosts
scp /etc/hosts/ root@10.10.12.130:/etc/hosts
scp ~/.bashrc root@datanode1:~/.bashrc
scp ~/.bashrc root@datanode2:~/.bashrc
```

3.3 配置 slaves 文件

登陆 master 节点，编辑 slaves 文件

```
cd $HADOOP_CONFIG_HOME/
vi slaves
```

加入以下信息

```
datanode1
datanode2
```

3.4 生成访问密钥

分别登陆 master, datanode1, datanode2 节点，输入：

```
ssh-keygen -t rsa
```

中途采用默认值（三次回车），结果如下：

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoop/.ssh/id_rsa):
Created directory '/home/hadoop/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoop/.ssh/id_rsa.
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub.
The key fingerprint is:
77:5d:01:9b:29:90:32:e8:79:e2:56:ac:a6:4c:2a:f2 hadoop@namenode
The key's randomart image is:
+--[ RSA 2048 ]-----+
  .  o  . . . . .
   .  o  o  . + .
    + +  . . . .
   . = S . . . .
    =
   + +
  o. o
  o. E
+-----+
[hadoop@namenode ~]$
```

登陆 datanode1 节点，输入：

```
scp /root/.ssh/id_rsa.pub root@master:/root/.ssh/id_rsa.pub.datanode1
```

登陆 datanode2 节点，输入：

```
scp /root/.ssh/id_rsa.pub root@master:/root/.ssh/id_rsa.pub.datanode2
```

登陆 master 节点，输入：

```
cd /root/.ssh
cat id_rsa.pub >> authorized_keys
cat id_rsa.pub.datanode1 >> authorized_keys
cat id_rsa.pub.datanode2 >> authorized_keys
chmod 664 ~/.ssh/authorized_keys
scp ~/.ssh/authorized_keys root@datanode1:/root/.ssh/authorized_keys
scp ~/.ssh/authorized_keys root@datanode2:/root/.ssh/authorized_keys
```

3.5 测试连通性：

```
[root@master hadoop]# ssh datanode1
Last login: Tue Mar 24 17:29:36 2015 from 10.10.0.2
[root@datanode1 ~]# exit
鎧海經
Connection to datanode1 closed.
[root@master hadoop]# ssh datanode2
Last login: Tue Mar 24 12:16:38 2015 from 10.10.0.2
[root@datanode2 ~]#
```

4、启动 hadoop

4.1 配置 datanode1, datanode2 节点

将 Java jdk 与 hadoop 写入另外两台机器:

Datanode1 节点:

```
scp -r /usr/lib/jvm/java root@datanode1:/usr/lib/jvm/  
scp -r /root/soft/ root@datanode1:/root/
```

Datanode2 节点:

```
scp -r /usr/lib/jvm/java root@datanode2:/usr/lib/jvm/  
scp -r /root/soft/ root@datanode2:/root/
```

4.2 启动 Hadoop

(1) 分别登陆 master, datanode1, datanode2 节点, 进入 hadoop 目录, 输入 start-all.sh 命令。

```
cd ~/soft/apache/hadoop/hadoop-2.6.0/etc/hadoop  
start-all.sh
```

结果如下:

```
[root@master hadoop-2.6.0]# cd etc/hadoop  
[root@master hadoop]# start-all.sh  
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh  
Starting namenodes on [master]  
master: namenode running as process 347. Stop it first.  
datanode1: datanode running as process 31728. Stop it first.  
datanode2: datanode running as process 6039. Stop it first.  
Starting secondary namenodes [0.0.0.0]  
0.0.0.0: secondarynamenode running as process 560. Stop it first.  
starting yarn daemons  
resourcemanager running as process 724. Stop it first.  
datanode1: starting nodemanager, logging to /root/soft/apache/hadoop/hadoop-2.6.  
0/logs/yarn-root-nodemanager-datanode1.out  
datanode2: starting nodemanager, logging to /root/soft/apache/hadoop/hadoop-2.6.  
0/logs/yarn-root-nodemanager-datanode2.out
```

(2) 使用 `hdfs dfsadmin -report` 查看主节点中的 hadoop 是否正常启动

```
cd ~/soft/apache/hadoop/hadoop-2.6.0/etc/hadoop  
hdfs dfsadmin -report
```

结果如下: 没有报错则 hadoopq 启动成功

```
[root@master hadoop]# cd ~/soft/apache/hadoop/hadoop-2.6.0/etc/hadoop
[root@master hadoop]# hdfs dfsadmin -report
Configured Capacity: 321962127360 (299.85 GB)
Present Capacity: 309726027776 (288.45 GB)
DFS Remaining: 309725986816 (288.45 GB)
DFS Used: 40960 (40 KB)
DFS Used%: 0.00%
Under replicated blocks: 0
Blocks with corrupt replicas: 0
Missing blocks: 0

-----
Live datanodes (3):
Name: 10.10.12.130:50010 (datanode2)
Hostname: datanode2
Decommission Status : Normal
Configured Capacity: 107320709120 (99.95 GB)
```