

HDFS8 新增节点与删除节点

当数据量越来越大的时候，原来有的数据节点的容量已经不能够满足存储数据的要求，需要在原有的集群基础上动态加多一个节点。

新的虚拟机操作

- 复制一台新的虚拟机

将我们纯净的虚拟机复制一台，作为我们新的数据节点

- 修改mac地址以及IP地址

```
# 因为克隆虚拟的默认的eth0网卡没了，出现的eth1没有ip配置
# 此时需要获取到虚拟机的MAC地址，选择虚拟机 设置->网络适配器->高级

#修改mac地址
vim /etc/udev/rules.d/70-persistent-net.rules

#修改ifcfg-eth0文件
vim /etc/sysconfig/network-scripts/ifcfg-eth0

添加或者修改 HWADDR=你的mac地址
```

- 重启系统

```
reboot
```

- 关闭防火墙

```
关闭防火墙
# service iptables stop

关闭selinux
# vim /etc/selinux/config
```

- 更改主机名

```
更改主机名为node04
# vim /etc/sysconfig/network
```

- 四台机器更改主机名 node3 与IP地址映射

```
注意每个节点都要配置
vim /etc/hosts

重启
reboot
```

- 生成公钥与私钥

```
执行以下命令生成公钥与私钥
# ssh-keygen
```

```
node03执行以下命令将node03的私钥拷贝到node01(master)服务器
# ssh-copy-id node1
```

```
node01执行以下命令将authorized_keys拷贝给node03
# cd /root/.ssh/
# scp authorized_keys node03:$PWD
```

- node3 安装 jdk

```
# 用整个集群的统一路径
# 复制其他节点的java文件并且配置好环境
```

- 解压Hadoop安装包

```
node01执行以下命令将Hadoop安装包拷贝到node03服务器
# cd /usr/local
# scp hadoop-2.7.2.tar.gz node03:$PWD
```

- 将 node1 关于 Hadoop 的配置文件全部拷贝到 node3

```
cd /usr/local/hadoop-2.7.2/etc/hadoop/scp ./ * node04:$PWD
```

服役新节点具体

- 创建 dfs.hosts 文件

在node01也就是namenode所在的机器的/export/servers/hadoop-2.6.0-cdh5.14.0/etc/hadoop目录下创建dfs.hosts文件

```
# cd /usr/local/hadoop-2.7.2/etc/hadoop
# touch dfs.hosts
# vim dfs.hosts
```

- 添加主机名称

```
node01
node02
node03
```

- node01 编辑 hdfs-site.xml 添加以下配置

在namenode的hdfs-site.xml配置文件中增加dfs.hosts属性

```
# cd /usr/local/hadoop-2.7.2/etc/Hadoop
# vim hdfs-site.xml
```

```
<property>
  <name>dfs.hosts</name>
  <value>usr/local/hadoop-2.7.2/etc/hadoop/dfs.hosts</value>
</property>
```

- 刷新 namenode

```
node01执行以下命令刷新namenode
# hdfs dfsadmin -refreshNodes
Refresh nodes successful
```

- 更新 resourceManager 节点

```
# yarn rmadmin -refreshNodes
19/03/16 11:19:47 INFO client.RMProxy: Connecting to ResourceManager at
node01/192.168.52.100:8033
```

- namenode 的 slaves 文件增加新服务节点主机名称

```
node01编辑slaves文件，并添加新增节点的主机，更改完后，slaves文件不需要分发到其他机器上面去
node01执行以下命令编辑slaves文件，slave 文件中记录的节点在集群启动时会进行启动。
# cd /usr/local/Hadoop-2.7.2/etc/hadoop
# vim slaves

node01
node02
node03
```

- 单独启动新增节点

```
在node03服务器执行以下命令，启动datanode和nodemanager
# cd /usr/local/hadoop-2.7.2/
# sbin/hadoop-daemon.sh start datanode      # 启动datanode
# sbin/yarn-daemon.sh start nodemanager      # 启动nodemanager
```

- 浏览器查看

<http://xxxx:50070/dfshealth.html#tab-overview>

<http://xxxx:8088/cluster>

- 使用负载均衡命令，让数据均匀负载所有机器

```
在node01上执行以下命令：
# cd /usr/local/hadoop-2.7.2/
# sbin/start-balancer.sh
```

退役旧数据节点

- 创建 dfs.hosts.exclude 配置文件

在namenode的cd /usr/local/hadoop-2.7.2/etc/hadoop目录下创建dfs.hosts.exclude文件，并添加需要退役的主机名称

node01执行以下命令

```
# cd /usr/local/hadoop-2.7.2/etc/hadoop
# touch dfs.hosts.exclude
# vim dfs.hosts.exclude
```

node3

- 编辑namenode所在机器的hdfs-site.xml

编辑namenode所在的机器的hdfs-site.xml配置文件，添加以下配置

node01执行以下命令

```
# cd /usr/local/hadoop-2.7.2/etc/hadoop
# vim hdfs-site.xml
```

```
<property>
  <name>dfs.hosts.exclude</name>
  <value>/usr/local/hadoop-2.7.2/etc/hadoop/dfs.hosts.exclude</value>
</property>
```

- 刷新namenode，刷新resourceManager

在namenode所在的机器执行以下命令，刷新namenode，刷新resourceManager

```
# hdfs dfsadmin -refreshNodes
# yarn rmadmin -refreshNodes
```

- 查看web浏览界面

<http://xxxx:50070/dfshealth.html#tab-datanode>

- 节点退役完成，停止该节点进程

等待退役节点状态为decommissioned（所有块已经复制完成），停止该节点及节点资源管理器。注意：如果副本数是3，服役的节点小于等于3，是不能退役成功的，需要修改副本数后才能退役。

node01 执行以下命令，停止该节点进程

```
# cd /usr/local/hadoop-2.7.2
# sbin/hadoop-daemon.sh stop datanode
# sbin/yarn-daemon.sh stop nodemanager
```

- 从include文件中删除退役节点

```
namenode所在节点也就是node01执行以下命令删除退役节点
# cd /usr/local/hadoop-2.7.2/etc/hadoop
# vim dfs.hosts

node01
node02

namenode所在节点也就是node01执行以下命令刷新namenode和resourceManager
# hdfs dfsadmin -refreshNodes
# yarn rmadmin -refreshNodes
```

- 从 namenode 的 slave 文件中删除退役节点

```
namenode所在机器也就是node01执行以下命令从slaves文件中删除退役节点

# cd /usr/local/hadoop-2.7.2/etc/hadoop
# vim slaves

node01
node02
```

- 如果数据负载不均衡，执行以下命令进行均衡负载

```
node01执行以下命令进行均衡负载
# cd /usr/local/hadoop-2.7.2/
# sbin/start-balancer.sh
```

- 打开浏览器检查

<http://xxxx:50070/dfshealth.html#tab-overview>

<http://xxxx:8088/cluster>