1. 搭建普通Hadoop集群(包含Zookeeper)到修改配置文件步骤
2. 修改文件：

core-site.xml

<configuration>

<!--指定HDFS的nameservice为ns1-->

<property>

<name>fs.defaultFS</name>

<value>hdfs://ns1</value>

</property>

<!--指定Hadoop临时目录-->

<property>

<name>hadoop.tmp.dir</name>

<value>/export/servers/hadoop-2.7.4/tmp</value>

</property>

<!--指定Zookeeper地址-->

<property>

<name>ha.zookeeper.quorum</name>

<value>node-01:2181,node-02:2181,node-03:2181</value>

</property>

</configuration>

hdfs-site.xml

<configuration>

<!--设置副本个数-->

<property>

<name>dfs.replication</name>

<value>2</value>

</property>

<!--设置namenode.name目录-->

<property>

<name>dfs.namenode.name.dir</name>

<value>file:/export/data/hadoop/name</value>

</property>

<!--设置namenode.data目录-->

<property>

<name>dfs.datanode.data.dir</name>

<value>file:/export/data/hadoop/data</value>

</property>

<!--开启WebHDFS-->

<property>

<name>dfs.webhdfs.enabled</name>

<value>true</value>

<!--在NN和DN上开启webHDFS（REST API）功能，不是必须-->

</property>

<!--指定HDFS的那么nameservice为ns1,需要和core-site.xml中的保持一致-->

<property>

<name>dfs.nameservices</name>

<value>ns1</value>

</property>

<!--ns1下面有两个namenode，分别是nn1和nn2-->

<property>

<name>dfs.ha.namenodes.ns1</name>

<value>nn1,nn2</value>

</property>

<!--nn1的RPC通信地址-->

<property>

<name>dfs.namenode.rpc-address.ns1.nn1</name>

<value>node-01:9000</value>

</property>

<!--nn1的http通信地址-->

<property>

<name>dfs.namenode.http-address.ns1.nn1</name>

<value>node-01:50070</value>

</property>

<!--nn2的RPC通信地址-->

<property>

<name>dfs.namenode.rpc-address.ns1.nn2</name>

<value>node-02:9000</value>

</property>

<!--nn2的http通信地址-->

<property>

<name>dfs.namenode.http-address.ns1.nn2</name>

<value>node-02:50070</value>

</property>

<!--指定namenode的元数据在JournalNode上的存放位置-->

<property>

<name>dfs.namenode.shared.edits.dir</name>

<value>qjournal://node-01:8485;node-02:8485;node-03:8485/ns1</value>

</property>

<!--指定JournalNode在本地磁盘存放数据的位置-->

<property>

<name>dfs.journalnode.edits.dir</name>

<value>/export/data/hadoop/journaldata</value>

</property>

<!--开启namenode失败自动切换-->

<property>

<name>dfs.ha.automatic-failover.enabled</name>

<value>true</value>

</property>

<!--配置失败自启动切换实现方式-->

<property>

<name>dfs.client.failover.proxy.provider.ns1</name>

<value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider</value>

</property>

<!--配置隔离机制方法，多个机制用换行分割，即每个机制暂用一行-->

<property>

<name>dfs.ha.fencing.methods</name>

<value>

sshfence

shell(/bin/true)

</value>

</property>

<!--使用sshfence隔离机制时需要ssh免登录-->

<property>

<name>dfs.ha.fencing.ssh.private-key-files</name>

<value>/root/.ssh/id\_rsa</value>

</property>

<!--配置sshfence隔离机制超时时间-->

<property>

<name>dfs.ha.fencing.ssh.connect-timeout</name>

<value>30000</value>

</property>

<property>

<name>dfs.ha.fencing.methods</name>

<value>sshfence</value>

</property>

</configuration>

mapred-site.xml

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

</configuration>

yarn-site.xml

<configuration>

<property>

<name>yarn.nodemanager.resource.memory-mb</name>

<value>2048</value>

</property>

<property>

<name>yarn.scheduler.maximum-allocation-mb</name>

<value>2048</value>

</property>

<property>

<name>yarn.nodemanager.resource.cpu-vcores</name>

<value>1</value>

</property>

<property>

<name>yarn.resourcemanager.ha.enabled</name>

<value>true</value>

</property>

<!--指定ResourceManager的cluster id-->

<property>

<name>yarn.resourcemanager.cluster-id</name>

<value>yrc</value>

</property>

<!--指定ResourceManager的名字-->

<property>

<name>yarn.resourcemanager.ha.rm-ids</name>

<value>rm1,rm2</value>

</property>

<!--分别指定ResourceManager的地址-->

<property>

<name>yarn.resourcemanager.hostname.rm1</name>

<value>node-01</value>

</property>

<property>

<name>yarn.resourcemanager.hostname.rm2</name>

<value>node-02</value>

</property>

<!--指定Zookeeper集群的地址-->

<property>

<name>yarn.resourcemanager.zk-address</name>

<value>node-01:2181,node-02:2181,node-03:2181</value>

</property>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

</configuration>

salves

node-01

node-02

node-03

hadoop-env.sh

export JAVA\_HOME=/export/servers/jdk

1. 启用hadoop高可用集群

cd /export/servers/zookeeper-3.4.10/bin

1. 启动集群各个节点的zookeeper服务

zkServer.sh start

1. 启动集群各个节点监控NameNode的管理日志的JournalNode

hadoop-daemon.sh start journalnode

1. 在node-01节点格式化NameNode

hadoop namenode -format

1. 将格式化后的目录复制到node-02和03上

scp -r /export/data/hadoop node-02:/export/data

scp -r /export/data/hadoop node-02:/export/data

1. 在node-01节点格式化ZKFC

hdfs zkfc -formatZK

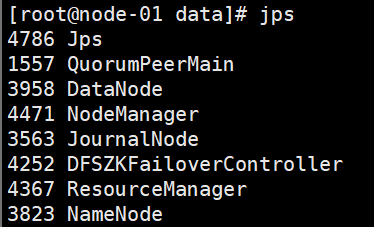
1. 在node-01节点启动HDFS

start-dfs.sh

1. 在node-01节点启动YARN

start-yarn.sh

jps查看node-01



(8个)

最后通过宿主机浏览器分别访问http://ip地址:50070和http://ip地址:8088查看HDFS和YARN集群状态

