**Configuring Hadoop High Availability Cluster(Two Namenode service)**

Configuring High Availability Hadoop Cluster:(Qjournal Method)

Steps:

Step1: Download and configure Zookeeper

Step2: Hadoop configuration and high availability settings

Step3: Creating folders for Hadoop cluster and file permissions

Step4: Hdfs service and file system format

Steps in Details:

Step 1: Download and configure Zookeeper

1.1 Download and configure Zookeeper software package from (https://www.apache.org/dist/zookeeper/zookeeper-3.4.5/zookeeper-3.4.5.tar.gz)

[cluster@n3:~]$wget https://www.apache.org/dist/zookeeper/zookeeper-3.4.5/zookeeper-3.4.5.tar.gz

Extract source

[cluster@n3:~]$tar –zxvf zookeeper-3.4.5.tar.gz

1.2 Zookeeper related configuration files are located

Configuration files: /home/cluster/zookeeper-3.4.5/conf

Binary executables: /home/cluster/zookeeper-3.4.5/bin

The Main configuration file

/home/cluster/zookeeper-3.4.5/conf/zoo.cfg

[cluster@n3:~]$cp zoo\_sample.cfg zoo.cfg

Modifying zoo.cfg as per our installation guide

[cluster@n3:~]$nano /home/cluster/zookeeper-3.4.5/conf/zoo.cfg

tickTime=2000

clientPort=3000

initLimit=5

syncLimit=2

dataDir=/home/cluster/zookeeper/data/

dataLogDir=/home/cluster/zookeeper/log/

server.1=n3:2888:3888

server.2=n4:2889:3889

Save & Exit!

Note :-If Each of the servers hosted in the same physical machine as instance , every server port number has changed to n3:2888:3888 , n4: 2889:3889

1.3 Create the folder structure for Zookeeper data and logs as defined in zoo.cfg , repeat following step in all the nodes in the cluster (n3 & n4)

[cluster@n3:~]$mkdir –p /home/cluster/zookeeper/data/

[cluster@n3:~]$mkdir –p /home/cluster/zookeeper/log/

1.4 Create the myid file in /home/cluster/zookeeper/data/ and assign the value of each of the nodes in cluster. (n3=1 & n4=2)

[cluster@n3:~]$nano /home/cluster/zookeeper/data/myid

1

Save and Exit!

[cluster@n4~]$nano /home/cluster/zookeeper/data/myid

2

Save & Exit!

Step 2: Hadoop configuration and high availability settings

Download the latest hadoop 2.X.X version from the releases

2.1 Add / modify the following lines in hadoop-env.sh file to apply environment variable settings.

[cluster@n3:~]$ nano /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/etc/hadoop/hadoop-env.sh

export JAVA\_HOME=/usr/lib/jvm/jdk1.7.0\_45/

export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=/home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/lib/native/

export HADOOP\_OPTS=”-Djava.library.path=/home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/lib/native/”

2.2 Add following lines in cores-site.xml file to configure journaling, default FS, temp directory & hdfs cluster. Within the <configuration> tag

[cluster@n3:~]$nano /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/etc/hadoop/core-site.xml

<property>

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

<value>/hdfs/dfs/tmp</value>

</property>

<property>

<name>fs.defaultFS</name>

<value>hdfs://mycluster</value>

</property>

<property>

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

<value>/hdfs/dfs/journal/data</value>

</property>

2.3 Add following lines in hdfs-site.xml file to configure dfs nameservice, dfs high availability, zookeeper & failover. Within the <configuration> tag.

[cluster@n3:~]$nano /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/etc/hadoop/hdfs-site.xml

<property>

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

<value>/hdfs/dfs/nn</value>

</property>

<property>

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

<value>/hdfs/dfs/dn</value>

</property>

<property>

<name>dfs.nameservices</name>

<value>mycluster</value>

<final>true</final>

</property>

<property>

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

<value>n3,n4</value>

<final>true</final>

</property>

<property>

<name>dfs.namenode.rpc-address.mycluster.n3</name>

<value>n3:8020</value>

</property>

<property>

<name>dfs.namenode.http-address.mycluster.n3</name>

<value>n3:50070</value>

</property>

<property>

<name>dfs.namenode.secondaryhttp-address.mycluster.n3</name>

<value>n3:50090</value>

</property>

<property>

<name>dfs.namenode.rpc-address.mycluster.n4</name>

<value>n4:8020</value>

</property>

<property>

<name>dfs.namenode.http-address.mycluster.n4</name>

<value>n4:50070</value>

</property>

<property>

<name>dfs.namenode.secondaryhttp-address.mycluster.n4</name>

<value>n4:50090</value>

</property>

<property>

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

<value>qjournal://n3:8485;n4:8485/mycluster</value>

</property>

<property>

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

<value>true</value>

</property>

<property>

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

<value>n3:3000,n4:3000</value>

</property>

<property>

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

<value>sshfence</value>

</property>

<property>

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

<value>/home/cluster/.ssh/id\_rsa</value>

</property>

<property>

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

<value>30000</value>

</property>

<property>

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

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

</property>

2.4 Add datanodes in the slaves configuration file as shown below.

[cluster@n3:~]$nano /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/etc/hadoop/slaves

n3

n4

Save & Exit!

Step 3 : Creating folders for Hadoop cluster and set file permissions

3.1 Create folder structure for journalnode as defined in core-site.xml, repeat following step in all the cluster nodes (n3 & n4)

[cluster@n3:~]$mkdir –p /hdfs/dfs/journal/data

3.2 Create temp folder for hadoop cluster as defined in core-site.xml, repeat following step in all the cluster nodes (n3 & n4)

[cluster@n3:~]$mkdir -p /hdfs/dfs/tmp

3.3 Create datanode and namenode folder for hadoop cluster as defined in hdfs-site.xml, repeat following step in all the cluster nodes (n3 & n4)

[cluster@n3:~]$mkdir -p /hdfs/dfs/dn

[cluster@n3:~]$mkdir -p /hdfs/dfs/nn

3.4 Copy hadoop source and zookeeper source configured in n3 node to n4

Step 4: Hdfs service and file system format

4.1 Start zookeeper service, once in all the nodes in cluster used for zookeeper, repeat below step in all the cluster nodes running zookeeper (n3 & n4).

Go to zookepeer-3.4.5 Binary path i.e /home/cluster/zookeeper-3.4.5/bin, then execute below commands

[cluster@n3:~]$./zkServer.sh start

[cluster@n4~]$./zkServer.sh start

4.2 Format Zookeeper file system in n3

Go to hadoop home path i.e /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2, then execute below command

[cluster@n3:~]$bin/hdfs zkfc –formatZK

Before format start journalnode in all the cluster nodes (n3 & n4)

[cluster@n3:~]$sbin/hadoop-daemon.sh start journalnode

4.3 Format namenode in n3

Go to hadoop home path i.e /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2, then execute below command

[cluster@n3:~]$bin/hdfs namenode –format

4.4 Copy Meta data information to slave name node in our guide (n4), run below command in

n4 (slave).

Make sure that namenode service is running in master node(n3)….

Go to hadoop home path i.e /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2, then execute below command

[cluster@n3:~]$sbin/hadoop-daemon.sh start namenode

Then in n4,

Go to hadoop home path i.e /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2, then execute below command

[cluster@n4~]$bin/hdfs namenode -bootstrapStandby

Start hadoop service............

[cluster@n3:~]$cd /home/cluster/hadoop-2.2.0-cdh5.0.0-beta-2/sbin

./stop-all.sh

and start again.

./start-dfs.sh

run jps to check services running in n3 & n4

[cluster@n3 sbin]$ jps

17588 DFSZKFailoverController

16966 DataNode

24142 Jps

4268 QuorumPeerMain

16745 NameNode

17276 JournalNode

[cluster@n4 bin]$ jps

14357 DFSZKFailoverController

2369 QuorumPeerMain

13906 DataNode

23689 Jps

15458 NameNode

14112 JournalNode