第一步、首先准备好六台linux系统的机器。

第二步、初始化环境配置：

（1）关闭六台机器防火墙和设置好静态ip地址。

（2）设置六台机器的主机名称并对应其自己的ip地址。

具体为：

192.168.57.91 clown1

192.168.57.92 clown2

192.168.57.93 clown3

192.168.57.94 clown4

192.168.57.95 clown5

192.168.57.96 clown6

（3）在clown1上安装jdk并依次复制到另5台机器。

复制命令为：scp -r /root/clldev/ root@192.168.57.92:/root/

第三步、集群规划：

|  |  |  |  |
| --- | --- | --- | --- |
| 主机名 | IP | 安装的软件 | 运行的进程 |
| clown1 | 192.168.57.91 | jdk、hadoop | NameNode、DFSZKFailoverController |
| clown2 | 192.168.57.92 | jdk、hadoop | NameNode、DFSZKFailoverController |
| clown3 | 192.168.57.93 | jdk、hadoop | ResourceManager |
| clown4 | 192.168.57.94 | jdk、hadoop、zookeeper | DataNode、NodeManager、JournalNode、QuorumPeerMain |
| clown5 | 192.168.57.95 | jdk、hadoop、zookeeper | DataNode、NodeManager、JournalNode、QuorumPeerMain |
| clown6 | 192.168.57.96 | jdk、hadoop、zookeeper | DataNode、NodeManager、JournalNode、QuorumPeerMain |

说明：

在hadoop2.0中通常由两个NameNode组成，一个处于active状态，另一个处于standby状态。Active NameNode对外提供服务，而Standby NameNode则不对外提供服务，仅同步active namenode的状态，以便能够在它失败时快速进行切换。

hadoop2.0官方提供了两种HDFS HA的解决方案，一种是NFS，另一种是QJM。这里我们使用简单的QJM。在该方案中，主备NameNode之间通过一组JournalNode同步元数据信息，一条数据只要成功写入多数JournalNode即认为写入成功。通常配置奇数个JournalNode

这里还配置了一个zookeeper集群，用于ZKFC（DFSZKFailoverController）故障转移，当Active NameNode挂掉了，会自动切换Standby NameNode为standby状态

第四步、zookeeper集群安装

安装步骤：

1.安装配置zooekeeper集群

1.1解压

tar -zxvf zookeeper-3.4.5.tar.gz -C /itcast/

1.2修改配置

cd /itcast/zookeeper-3.4.5/conf/

cp zoo\_sample.cfg zoo.cfg

vim zoo.cfg

修改：dataDir=/itcast/zookeeper-3.4.5/tmp

在最后添加：

server.1=itcast04:2888:3888

server.2=itcast05:2888:3888

server.3=itcast06:2888:3888

保存退出

然后创建一个tmp文件夹

mkdir /itcast/zookeeper-3.4.5/tmp

再创建一个空文件

touch /itcast/zookeeper-3.4.5/tmp/myid

最后向该文件写入ID

echo 1 > /itcast/zookeeper-3.4.5/tmp/myid

1.3将配置好的zookeeper拷贝到其他节点(首先分别在itcast05、itcast06根目录下创建一个itcast目录：mkdir /itcast)

scp -r /itcast/zookeeper-3.4.5/ itcast05:/itcast/

scp -r /itcast/zookeeper-3.4.5/ itcast06:/itcast/

注意：修改itcast05、itcast06对应/itcast/zookeeper-3.4.5/tmp/myid内容

itcast05：

echo 2 > /itcast/zookeeper-3.4.5/tmp/myid

itcast06：

echo 3 > /itcast/zookeeper-3.4.5/tmp/myid

第五步、hadoop集群安装

2.安装配置hadoop集群

2.1解压

tar -zxvf hadoop-2.2.0.tar.gz -C /itcast/

2.2配置HDFS（hadoop2.0所有的配置文件都在$HADOOP\_HOME/etc/hadoop目录下）

#将hadoop添加到环境变量中

vim /etc/profile

export JAVA\_HOME=/usr/java/jdk1.7.0\_55

export HADOOP\_HOME=/itcast/hadoop-2.2.0

export PATH=$PATH:$JAVA\_HOME/bin:$HADOOP\_HOME/bin

#hadoop2.0的配置文件全部在$HADOOP\_HOME/etc/hadoop下

cd /itcast/hadoop-2.2.0/etc/hadoop

2.2.1修改hadoo-env.sh

export JAVA\_HOME=/usr/java/jdk1.7.0\_55

2.2.2修改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>/itcast/hadoop-2.2.0/tmp</value>

</property>

<!-- 指定zookeeper地址 -->

<property>

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

<value>itcast04:2181,itcast05:2181,itcast06:2181</value>

</property>

</configuration>

2.2.3修改hdfs-site.xml

<configuration>

<!--指定hdfs的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，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>itcast01:9000</value>

</property>

<!-- nn1的http通信地址 -->

<property>

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

<value>itcast01:50070</value>

</property>

<!-- nn2的RPC通信地址 -->

<property>

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

<value>itcast02:9000</value>

</property>

<!-- nn2的http通信地址 -->

<property>

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

<value>itcast02:50070</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://itcast04:8485;itcast05:8485;itcast06:8485/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>/itcast/hadoop-2.2.0/journal</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免登陆 -->

<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>

</configuration>

2.2.4修改mapred-site.xml

<configuration>

<!-- 指定mr框架为yarn方式 -->

<property>

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

<value>yarn</value>

</property>

</configuration>

2.2.5修改yarn-site.xml

<configuration>

<!-- 指定resourcemanager地址 -->

<property>

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

<value>itcast03</value>

</property>

<!-- 指定nodemanager启动时加载server的方式为shuffle server -->

<property>

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

<value>mapreduce\_shuffle</value>

</property>

</configuration>

2.2.6修改slaves(slaves是指定子节点的位置，因为要在itcast01上启动HDFS、在itcast03启动yarn，所以itcast01上的slaves文件指定的是datanode的位置，itcast03上的slaves文件指定的是nodemanager的位置)

itcast04

itcast05

itcast06

最后，将配置好的hadoop拷贝到其他节点

scp -r /itcast/ itcast02:/

scp -r /itcast/ itcast03:/

scp -r /itcast/hadoop-2.2.0/ root@itcast04:/itcast/

scp -r /itcast/hadoop-2.2.0/ root@itcast05:/itcast/

scp -r /itcast/hadoop-2.2.0/ root@itcast06:/itcast/

第六步、配置免密码登陆

#首先要配置itcast01到itcast02、itcast03、itcast04、itcast05、itcast06的免密码登陆

#在itcast01上生产一对钥匙

ssh-keygen -t rsa

#将公钥拷贝到其他节点，包括自己

ssh-coyp-id itcast01

ssh-coyp-id itcast02

ssh-coyp-id itcast03

ssh-coyp-id itcast04

ssh-coyp-id itcast05

ssh-coyp-id itcast06

#配置itcast03到itcast04、itcast05、itcast06的免密码登陆

#在itcast03上生产一对钥匙

ssh-keygen -t rsa

#将公钥拷贝到其他节点

ssh-coyp-id itcast04

ssh-coyp-id itcast05

ssh-coyp-id itcast06

#注意：两个namenode之间要配置ssh免密码登陆，别忘了配置itcast02到itcast01的免登陆

在itcast02上生产一对钥匙

ssh-keygen -t rsa

ssh-coyp-id -i itcast01

第七步、启动集群

###注意：严格按照下面的步骤

2.5启动zookeeper集群（分别在itcast04、itcast05、itcast06上启动zk）

cd /itcast/zookeeper-3.4.5/bin/

./zkServer.sh start

#查看状态：一个leader，两个follower

./zkServer.sh status

2.6启动journalnode（在itcast01上启动所有journalnode，注意：是调用的hadoop-daemons.sh这个脚本，注意是复数s的那个脚本）

cd /itcast/hadoop-2.2.0

sbin/hadoop-daemons.sh start journalnode

#运行jps命令检验，itcast04、itcast05、itcast06上多了JournalNode进程

2.7格式化HDFS

#在itcast01上执行命令:

hdfs namenode -format

#格式化后会在根据core-site.xml中的hadoop.tmp.dir配置生成个文件，这里我配置的是/itcast/hadoop-2.2.0/tmp，然后将/itcast/hadoop-2.2.0/tmp拷贝到itcast02的/itcast/hadoop-2.2.0/下。

scp -r tmp/ itcast02:/itcast/hadoop-2.2.0/

2.8格式化ZK(在itcast01上执行即可)

hdfs zkfc -formatZK

2.9启动HDFS(在itcast01上执行)

sbin/start-dfs.sh

2.10启动YARN(#####注意#####：是在itcast03上执行start-yarn.sh，把namenode和resourcemanager分开是因为性能问题，因为他们都要占用大量资源，所以把他们分开了，他们分开了就要分别在不同的机器上启动)

sbin/start-yarn.sh

到此，hadoop2.2.0配置完毕，可以统计浏览器访问:

http://192.168.1.201:50070

NameNode 'itcast01:9000' (active)

http://192.168.1.202:50070

NameNode 'itcast02:9000' (standby)

第八步、验证集群

验证HDFS HA

首先向hdfs上传一个文件

hadoop fs -put /etc/profile /profile

hadoop fs -ls /

然后再kill掉active的NameNode

kill -9 <pid of NN>

通过浏览器访问：http://192.168.1.202:50070

NameNode 'itcast02:9000' (active)

这个时候itcast02上的NameNode变成了active

在执行命令：

hadoop fs -ls /

-rw-r--r-- 3 root supergroup 1926 2014-02-06 15:36 /profile

刚才上传的文件依然存在！！！

手动启动那个挂掉的NameNode

sbin/hadoop-daemon.sh start namenode

通过浏览器访问：http://192.168.1.201:50070

NameNode 'itcast01:9000' (standby)

验证YARN：

运行一下hadoop提供的demo中的WordCount程序：

hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.2.0.jar wordcount /profile /out