# Hadoop+hbase+hive+kylin安装、搭建手册

* **软件版本**

1. Jdk1.7+
2. Zookeeper3.4.6
3. Hadoop2.6.0
4. Hbase1.1.5
5. Hive2.1.0
6. Kylin1.5.0

集群安装前，需要做如下准备：

1. 关闭防火墙

关闭命令：service iptables stop

永久关闭防火墙：chkconfig iptables off

查看防火墙状态：service iptables status

注：个人建议永久关闭防火墙，如果重启系统有时候会忘记关闭防火墙，导致集群连接失败。

1. 修改机器hostname及机器别名

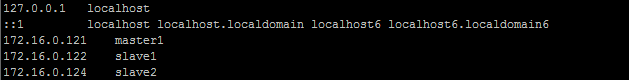
打开network配置文件：vi /etc/sysconfig/network

修改配置如下：



打开hosts配置文件：vi /etc/hosts

修改配置如下：



其中master1为主机，slave1\slave2为从机。

1. 建立机器互信，即ssh无密码登陆

准备工作：  
    1、确认本机sshd的配置文件（需要root权限）  
　　vi /etc/ssh/sshd\_config  
　　找到以下内容，并去掉注释符”#“  
　　RSAAuthentication yes  
　　PubkeyAuthentication yes  
　　AuthorizedKeysFile      .ssh/authorized\_keys  
　  2、如果修改了配置文件需要重启sshd服务 （需要root权限）  
　　/sbin/service sshd restart

配置SSH无密码登录需要3步：  
    1.生成公钥和私钥  
    2.导入公钥到认证文件,更改权限  
    3.测试  
  
  
1.生成公钥和私钥  
  Shell代码:  ssh-keygen -t rsa

ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa  
   默认在 ~/.ssh目录生成两个文件：  
    id\_rsa      ：私钥  
    id\_rsa.pub  ：公钥  
2.导入公钥到认证文件,更改权限  
    2.1 导入本机  
        Shell代码:  cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys    
    2.2 导入要免密码登录的服务器  
        首先将公钥复制到服务器  
Shell代码:scp ~/.ssh/id\_rsa.pub xxx@host:/home/id\_rsa.pub    
   然后，将公钥导入到认证文件(这一步的操作在服务器上进行)  
  Shell代码:  cat /home/id\_rsa.pub >> ~/.ssh/authorized\_keys   
    2.3 在服务器上更改权限  
     Shell代码: chmod 700 ~/.ssh chmod 600 ~/.ssh/authorized\_keys

* **安装mysql**

**查看是否安装了mysql（master1）**

ps -aux | grep mysql

* **安装jdk**

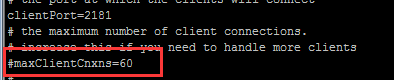
查看安装版本：java -version

未安装时，下载tar包解压后配置环境变量后检查。

* **Zookeeper**

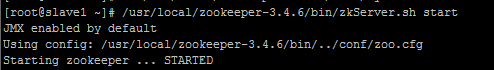
1. 安装
2. 解压缩zookeeper到根目录下，进入目录，创建文件夹data、datalog、logs， 配置环境变量

export ZOOKEEPER\_HOME=/zookeeper-3.4.6 export PATH=$PATH:$ZOOKEEPER\_HOME/bin

1. 进入conf文件夹，复制zoo\_sample.cfg 为zoo.cfg cp zoo\_sample.cfg zoo.cfg
2. vi zoo.cfg修改zoo.cfg，增加以下内容后保存并退出： dataDir=/zookeeper-3.4.6/data dataLogDir=/zookeeper-3.4.6/datalog server.1=master1:2888:3888 server.2=slave1:2888:3888 server.3=slave2:2888:3888  解开上面红框中内容的注释，并设置为300，该配置是设置zk的重连次数，默认次数为60.此处不解开有可能造成hbase连接失败。
3. 修改conf下得log4j.properties文件，配置log文件生成位置 zookeeper.log.dir=/zookeeper-3.4.6/logs zookeeper.log.file=zookeeper-3.4.6.log zookeeper.tracelog.dir=/zookeeper-3.4.6/logs zookeeper.tracelog.file=zookeeper\_trace.log
4. 进入data文件夹创建文件myid，添加内容1保存退出
5. 分发zookeeper文件夹到各个从机上的根目录上

Scp -r /zookeeper-3.4.6 root@slave1:/usr/local/ scp -r /zookeeper-3.4.6 root@slave2:/usr/local/

1. 修改每台机上的myid文件，按照master1的myid=1，slave1的myid=2依次类推。
2. 简单使用
3. 分别进入各个主机的zookeeper目录下启动zk服务 bin/zkServer.sh start 注：必须先启动主机zk服务，否则会造成从机myid文件出现变化，对排错造成错误判断。
4. 分别查询zk的状态 bin/zkServer.sh status Leader 是zk主机启动了 Follower是从机启动了
5. 停止zk bin/zkServer.sh stop
6. 常见问题及解决办法



Zk启动失败

1. 原因1：端口占用，导致程序启动失败。 解决办法：输入命令 netstat -nltp 查看2181端口使用情况， 1使用kill 【pid】释放端口 2 修改zk配置文件zoo.cfg中的端口号
2. 原因2：防火墙未关闭 解决办法：关闭防火墙。
3. 原因3：配置文件中参数或路径中有空格 解决办法：检查配置文件，并修改配置文件参数。

* **Hadoop**

1. 安装
2. 解压缩hadoop-2.6.0到根目录，进入目录，创建文件夹tmp、hdfs/name、hdfs/data
3. 进入hadoop/etc/hadoop，该文件夹包含了hadoop的大部分配置文件。
4. 修改hadoop各个配置文件

Core-site.xml

<configuration>

<property>

<name>fs.default.name</name>

<value>hdfs://master1:9000</value>

<final>true</final>

</property>

<property>

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

<value>/home/hadoop/hadoop/tmp</value>

<description>A base for other tempory directories</description>

</property>

<property>

<name>io.file.buffer.size</name>

<value>131702</value>

</property>

<property>

<name>fs.checkpoint.period</name>

<value>3600</value>

<description>多长时间记录一次hdfs的镜像，默认一小时</description>

</property>

<property>

<name>fs.checkpoint.size</name>

<value>67108864</value>

<description>一次记录多大的size，默认64M</description>

</property>

</configuration>

设置hadoop-env.sh

#设置JAVA\_HOME

export JAVA\_HOME=/jdk1.7.0\_80

#设置HADOOP\_CONF\_DIR

export HADOOP\_CONF\_DIR=${HADOOP\_CONF\_DIR:-"/etc/hadoop"}

设置hdfs-site.xml

<configuration>

<property>

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

<value>/home/hadoop/hadoop/hdfs/name</value>

</property>

<property>

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

<value>/home/hadoop/hadoop/hdfs/data</value>

</property>

<property>

<name>dfs.replication</name>

<value>4</value>

</property>

<property>

<name>dfs.namenode.secondary.http-address</name>

<value>master2:9001</value>

</property>

<property>

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

<value>true</value>

</property>

<property>

<name>dfs.client.read.shortcircuit</name>

<value>false</value>

</property>

</configuration>

设置mapred-site.xml

<configuration>

　　<property>

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

　　<value>yarn</value>

</property>

　　<property>

　　<name>mapreduce.jobtracker.http.address</name>

　 <value>NameNode:50030</value>

　　</property>

<property>

<name>mapreduce.jobhistory.address</name>

<value>master1:10020</value>

</property>

<property>

<name>mapreduce.jobhistory.webapp.address</name>

<value>master1:19888</value>

</property>

</configuration>

新增masters文件，以部署高可用的hadoop，将slave1和slave2作为secondary name node

Masters文件内容：master1

修改slaves文件，修改localhost为master1，新增slave1，slave2

修改yarn-env.sh

Export JAVA\_HOME=/jdk1.7.0\_67

修改yarn-site.xml

<configuration>

<property>

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

<value>master1:2181,master2:2181,slave1:2181,slave2:2181</value>

</property>

<property>

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

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

<property>

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

<value>master1:8032</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>master1:8030</value>

</property>

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>master1:8031</value>

</property>

<property>

<name>yarn.resourcemanager.admin.address</name>

<value>master1:8033</value>

</property>

<property>

<name>yarn.resourcemanager.webapp.address</name>

<value>master1:8088</value>

</property>

<property>

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

<value>2048</value>

</property>

</configuration>

配置hadoop环境变量

export HADOOP\_HOME=/home/hadoop/hadoop

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

1. 简单使用

查看hadoop是否安装配置成功

命令：Hadoop version

分发hadoop文件夹到各个机器上的根目录

Scp -r ~/hadoop root@slave1:/usr/local/

Scp -r ~/hadoop root@slave2:/usr/local/

重新格式化hdfs系统

如果集群刚配置从没启动过，直接执行格式化操作。

如果集群已经格式化了后启动过，则先执行删除旧数据的操作后，在执行格式化操作。

1）删除旧数据

在 hdfs-ste.xml 配置了

dfs.name.dir = /home/hadoop/hdfs/name (namenode上存储hdfs名字空间元数据)

dfs.data.dir = /home/hadoop/hdfs/data (datanode上数据块的物理存储位置)

在core-site.xml中配置了

hadoop.tmp.dir = /home/hadoop/hadoop/tmp（namenode上本地的hadoop临时文件夹）

将各个集群节点这三个文件夹下面的文件和目录全部删除

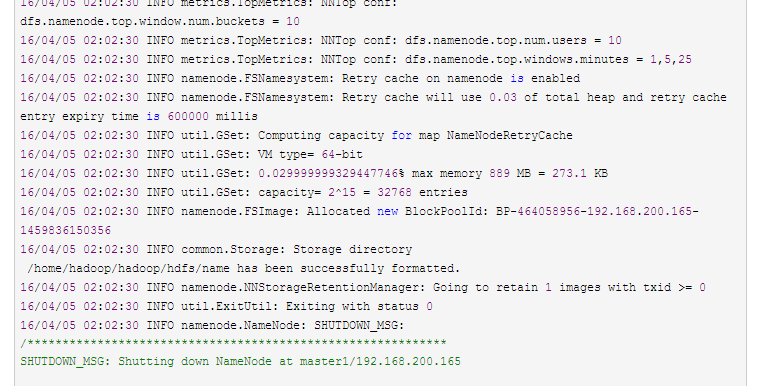
2）执行格式化命令（仅在主机格式化即可）

hadoop namenode -format

或者使用命令

hdfs namenode -format

格式化完成的标志：



无报错，与上图内容相似。

启动停止hadoop

注意事项：

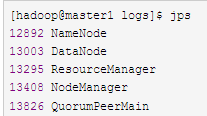
各个集群节点之间能免密码登录彼此

确保各个节点配置的myid和zoo.cfg里的一致，并先启动zookeeper集群

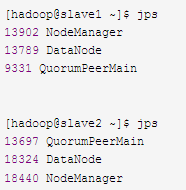
启动命令：sbin/start-all.sh或者依次执行start-dfs.sh和start-yarn.sh

启动完成后查看守护进程：命令：jps

主机：



从机：



停止hadoop集群

Sbin/stop-all.sh

启动成功后可以访问web控制台查看集群信息

NAMENODE：<http://192.168.200.165:50070/>

SECONDARY NAMENODE：[http://192.168.200.166:9001](http://192.168.200.166:9001/)

Nodes Of Cluster: [http://192.168.200.166:8088](http://192.168.200.166:8088/)

因为kylin中需要mapreduce任务调度，所以需要启动jobhistoryserver

mr-jobhistory-daemon.sh start historyserver

查看守护进程



1. 常见问题及解决办法

Namenode或者datanode进程无法启动

删除tmp和hdfs文件下得信息，重新格式化hadoop，重启hadoop集群。（暂时只碰到这一个问题，并使用该方法解决多次，多次搭建hadoop后发现，这个问题和防火墙有关，所以关闭防火墙后hadoop并不容易出现问题）

* **HBase**

1. 安装

解压缩hbase到根目录下，进入目录hbase

进入conf文件夹

修改hbase各个配置文件如下：

Hbase-env.sh

Export JAVA\_HOME=/jdk1.7.0\_67

Regionservers

Master1

Slave1

Slave2

Hbase-site.xml

<property>

<name>hbase.rootdir</name>

<value>hdfs://master1:9000/hbase</value>

</property>

<property>

<name>hbase.cluster.distributed</name>

<value>true</value>

</property>

<property>

<name>hbase.zookeeper.quorum</name>

<value>master1,master2,slave1,slave2</value>

</property>

<property>

<name>hbase.zookeeper.property.dataDir</name>

<value>/zookeeper-3.4.6/data</value>

</property>

<property>

<name>hbase.zookeeper.property.clientPort</name>

<value>2181</value>

</property>

<property>

<name>hbase.master.info.bindAddress</name>

<value>master1</value>

</property>

<property>

<name>hbase.master.info.port</name>

<value>60010</value>

</property>

配置环境变量，优化内存

export HBASE\_HOME=/home/hadoop/hbase

export PATH=$PATH:$HBASE\_HOME/bin

export HBASE\_MASTER\_OPTS="$HBASE\_MASTER\_OPTS -Xms1536m -Xmx2048m -Xmn1024m -XX:+UseParNewGC -XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=70

-XX:PermSize=512m -XX:MaxPermSize=512m"

export HBASE\_REGIONSERVER\_OPTS="$HBASE\_REGIONSERVER\_OPTS

-Xms1536m -Xmx2048m -Xmn1024m -XX:+UseParNewGC -XX:+UseConcMarkSweepGC-XX:CMSInitiatingOccupancyFraction=70 -XX:PermSize=512m -XX:MaxPermSize=512m"

export HBASE\_THRIFT\_OPTS="$HBASE\_THRIFT\_OPTS -Xms1024m -Xmx2048m"

分发hbase文件夹到各个机器上的根目录

scp -r ~/hbase/ hadoop@slave1:~

scp -r ~/hbase/ hadoop@slave2:~

1. 简单使用

启动和停止hbase

bin/start-hbase.sh

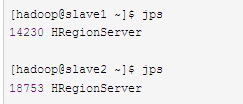
bin/stop-hbase.sh

启动成功后查看守护进程

Master1



从机

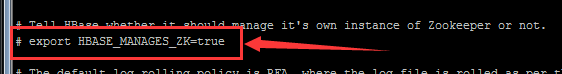


1. 常见问题及解决办法

问题1：地址已使用

原因：hbase默认会选择自带的zk，自带zk使用了2181端口

解决办法：



解开该出注释，并设置成false即可

问题2：hbase无法连接zk，导致hmaster进程自动断开

原因：在默认的重连次数内，hbase无法和zk建立心跳

解决办法：



解开该处注释，并设置成300，zk默认重连次数为60

问题3：集群节点的时间不同步

org.apache.hadoop.hbase.ClockOutOfSyncException

解决办法：设置时间同步

#yum install ntpdate

# ntpdate 0.asia.pool.ntp.org

#rm -rf /etc/localtime  
 #ln -s /usr/share/zoneinfo/Asia/Shanghai /etc/localtime

* **Hive**

1. 安装

解压缩hive到根目录下，进入目录hive

进入conf文件夹

在mysql上创建元数据库hive

注：在master1上创建数据库hive，编码选latin，如果不选，会导致hive启动过程中建表失败，因为hive建表使用的sql语句长多大部分为4000；

为hive数据库授权

grant all on hive.\* to 'root'@'%' IDENTIFIED BY 'weidong' with grant option;

flush privilege;

设置mysql数据库为任意IP可连接

update user set host='%' where host='localhost';

注：该设置可能引发hive连接不上的错误，注意使用。

修改hive各配置文件如下：

<configuration>

<property>

　　<name>javax.jdo.option.ConnectionURL</name>

　　　　<value>jdbc:mysql://master1:3306/hive?createDatabaseIfNotExist=true</value>

</property>

<property>

<name>javax.jdo.option.ConnectionDriverName</name>

<value>com.mysql.jdbc.Driver</value>

<description>驱动名</description>

</property>

<property>

<name>javax.jdo.option.ConnectionUserName</name>

<value>root</value>

<description>用户名</description>

</property>

<property>

<name>javax.jdo.option.ConnectionPassword</name>

<value>weidong</value>

<description>密码</description>

</property>

<property>

<name>datanucleus.schema.autoCreateTables</name>

<value>true</value>

</property>

<property>

<name>hive.metastore.warehouse.dir</name>

<value>hdfs://master1:9000/home/hadoop/hive/warehouse</value>

<description>数据路径（相对hdfs）</description>

</property>

<property>

<name>hive.exec.scratchdir</name>

<value>hdfs://master1:9000/home/hadoop/hive/warehouse</value>

</property>

<property>

<name>hive.querylog.location</name>

<value>/home/hadoop/hive/logs</value>

</property>

<property>

<name>hive.aux.jars.path</name>

<value>file:///home/hadoop/hbase/lib</value>

</property>

<property>

<name>hive.metastore.uris</name>

<value>thrift://master1:9083</value>

<description>运行hive得主机地址及端口</description>

</property>

</configuration>

将日志配置打开

cp hive-log4j2.properties.template hive-log4j2.properties

cp hive-exec-log4j2.properties.template hive-exec-log4j2.properties

1. 简单使用

启动hive

首先需要启动元数据库

hive --service metastore &

启动hive客户端

使用show tables查看hive下是否有表；

停止hive

Ps -aux | grep hive 查看hive目前的进程pid，用kill干掉即可

1. 常见问题及解决办法

错误1：找不到驱动导致数据库连接失败

原因：hive初始化过程中缺少，数据库驱动包支持

解决办法：导入数据库驱动包到/hive/lib目录下

* **Kylin**

1. 安装

了解kylin的两种二进制包

预打包的二进制安装包：[apache-kylin-1.5.0-bin.tar.gz](https://dist.apache.org/repos/dist/release/kylin/apache-kylin-1.5.0/apache-kylin-1.5.0-bin.tar.gz)

特别二进制包：[apache-kylin-1.5.0-HBase1.1.3-bin.tar.gz](https://dist.apache.org/repos/dist/release/kylin/apache-kylin-1.5.0/apache-kylin-1.5.0-HBase1.1.3-bin.tar.gz)

注：选包过程中尽量选择特别二进制包。否则kylin在执行job过程中会引发部分错误，倒是构建cube失败。

解压缩kylin到根目录下，进入目录kylin

在/etc/profile里配置kylin环境变量和一个名为hive\_dependency的变量

export KYLIN\_HOME=/home/hadoop/kylin

export PATH=$PATH:$ KYLIN\_HOME/bin

export hive\_dependency=/home/hadoop/hive/conf:/home/hadoop/hive/lib/\*:/home/hadoop/hive/hcatalog/share/hcatalog/hive-hcatalog-core-2.0.0.jar

修改kylin的启动脚本kylin.sh

1）显式声明 KYLIN\_HOME

export KYLIN\_HOME=/home/Hadoop/kylin

2）在HBASE\_CLASSPATH\_PREFIX中显示增加$hive\_dependency依赖

export HBASE\_CLASSPATH\_PREFIX=${tomcat\_root}/bin/bootstrap.jar:${tomcat\_root}/bin/tomcat-juli.jar:${tomcat\_root}/lib/\*:$hive\_dependency:$HBASE\_CLASSPATH\_PREFIX

检查环境是否设置成功



进入conf文件夹，修改kylin各个配置文件如下

Kylin.properties

kylin.owner=wdcloud@kylin.apache.org

kylin.rest.servers=master1:7070

kylin.hdfs.working.dir=/home/hadoop/kylin/kylin\_hdfs\_working\_dir

kylin.job.remote.cli.working.dir=/home/hadoop/kylin/kylin\_job\_working\_dir

#定义kylin用于MR jobs的job.jar包和hbase的协处理jar包，用于提升性能。

kylin.job.jar=/home/hadoop/kylin/lib/kylin-job-1.5.0-SNAPSHOT.jar

kylin.coprocessor.local.jar=/home/hadoop/kylin/lib/kylin-coprocessor-1.5.0-SNAPSHOT.jar

注：hadoop未安装snappy压缩工具时，使用kylin需要修改配置



将该配置置空，否则会导致kylin构建cube失败，hadoop会抛出找不到压缩包的异常。

将kylin\_hive\_conf.xml和kylin\_job\_conf.xml的副本数设置为4

<property>

<name>dfs.replication</name>

<value>3</value>

<description>Block replication</description>

</property>

注：hadoop未安装snappy压缩工具时，以上两个配置文件中包含comprss的配字节注释掉。异常同上。

1. 简单使用

启动和停止kylin

#确认必须启动的服务：

1)hadoop2的hdfs/yarn/jobhistory服务

    start-dfs.sh

    start-yarn.sh

    mr-jobhistory-daemon.sh start historyserver

2)hive 元数据库：hive --service metastore &

3)zookeeper

4)hbase ：start-hbase.sh

#检查hive和hbase的依赖

[hadoop@master1 kylin]$ find-hive-dependency.sh

[hadoop@master1 kylin]$ find-hbase-dependency.sh

#启动和停止kylin的命令如下：

[hadoop@master1 kylin]$ kylin.sh start

[hadoop@master1 kylin]$ kylin.sh stop

Web访问地址

<http://192.168.200.165:7070/kylin/login>

默认的登录username/password 是 ADMIN/KYLIN.

1. 常见问题及解决办法

* **常用命令**

**linux常用命令：**

**Cat /usr/local/\*\*\* 查看文件**

**Ls /usr/ 查看目录下文件名**

**Ll 查看目录下文件详细信息**

**Rm -rf 强制删除文件夹和文件夹下所有内容**

**Rm 删除文件**

**Mkdir 创建文件夹**

**Vi 编写文件**

**Source /etc/profile 使环境变量立即生效**

**Service iptables stop|start 关闭|打开防火墙**

**Service network restart 重启网络服务**

**Service mysqld start|stop|restart 启动|关闭|重启 mysql服务**

**Tar zxvf 解压gz压缩包**

**Scp -r \*\*\*\* root@slave1：/\*\*\*\* 拷贝文件到其他机器**

**Ssh --机器名称 使用ssh登陆到其他机器**

**Zookeeper常用命令：**

**Bin/zkServer.sh start|stop|status 启动|停止|查看 zk服务**

**Hadoop 常用命令：**

**Hbase 常用命令：**

**Hive 常用命令：**

**Kylin 常用命令：**