

Hive1 安装与环境配置

准备好的安装包：

1. apache-hive-2.3.6-bin.tar.gz
2. mysql-8.0.17-linux-glibc2.12-x86_64.tar.xz
3. mysql-connector-java-8.0.21.jar

安装前说明：

1. 安装hive前提是要先**安装**hadoop集群
2. hive只需要再hadoop的**namenode节点**集群里安装即可(需要在所有namenode上安装), 可以不在datanode节点的机器上安装。
3. 虽然修改配置文件并不需要你已经把hadoop跑起来, 但是本文中用到了hadoop命令, 在执行这些命令前你必须确保hadoop是在正常跑着的, 而且启动hive的前提也是需要hadoop在正常跑着, 所以建议先将hadoop**搭建起来并且运行**。

第一步：mysql安装

因为Hive的默认元数据是mysql, 所以必须要先安装Mysql

yum安装

```
#首先查看mysql是否已经安装:
rpm -qa | grep mysql

#如果已经安装, 想删除的话
rpm -e mysql #普通删除命令
rpm -e --nodeps mysql #强力删除命令

#如果没有, 则通过wget命令下载该包
wget http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm

#下载成功后, 输入命令安装, 直接y即可
yum install mysql-server

#安装完成, 启动服务
service mysqld start

#设置账号密码, 输入之后直接回车
mysqladmin -u root -p password '123456'

#然后输入密码进入mysql
mysql -u root -p

#通过授权法更改远程连接权限
grant all privileges on . to 'root'@'%' identified by '123456';
注: 第一个'root'是用户名, 第二个'%'是所有的ip都可以远程访问, 第三个'123456'表示 用户密码 如果不常用 就关掉
```

```
#刷新
flush privileges;
```

编译包安装

```
# 解压
tar -xvf mysql-8.0.17-linux-glibc2.12-x86_64.tar.xz

#移动
mv mysql-8.0.17-linux-glibc2.12-x86_64 /usr/local

#改名
cd /usr/local
mv mysql-8.0.17-linux-glibc2.12-x86_64 mysql

#安装mysql
cd /usr/local/mysql
./scripts/mysql_install_db --user=mysql

#启动mysql
service mysql start

#设置账号密码,输入之后直接回车
mysqladmin -u root -p password '123456'

#然后输入密码进入mysql
mysql -u root -p

#通过授权法更改远程连接权限
grant all privileges on . to 'root'@%' identified by '123456';
注:第一个'root'是用户名,第二个'%'是所有的ip都可以远程访问,第三个'123456'表示 用户密码 如果不常用 就关闭掉

#刷新
flush privileges;
```

Hive环境安装和配置

文件准备

```
#解压
tar -xvf apache-hive-2.3.6-bin.tar.gz

#改名
mv apache-hive-2.3.6-bin /opt/hive
mv apache-hive-2.3.6-bin hive2.3
```

环境配置

```
#编辑 /etc/profile 文件
vim/etc/profile
添加:
export HIVE_HOME=/opt/hive/hive2.3
export HIVE_CONF_DIR=${HIVE_HOME}/conf
export
PATH=.:${JAVA_HOME}/bin:${SCALA_HOME}/bin:${SPARK_HOME}/bin:${HADOOP_HOME}/bin:${ZK_HOME}/bin:${HBASE_HOME}/bin:${HIVE_HOME}/bin:$PATH

#输入
source /etc/profile
```

配置更改

1. 创建文件夹

```
#在修改配置文件之前，需要先在root目录下建立一些文件夹。
mkdir /root/hive
mkdir /root/hive/warehouse

#新建完该文件之后，需要让hadoop新建/root/hive/warehouse 和 /root/hive/ 目录。
hadoop fs -mkdir -p /root/hive/
hadoop fs -mkdir -p /root/hive/warehouse

#给刚才新建的目录赋予读写权限，执行命令：
hadoop fs -chmod 777 /root/hive/
hadoop fs -chmod 777 /root/hive/warehouse

#检擦是否创建成功
hadoop fs -ls /root/
hadoop fs -ls /root/hive/
```

2. 修改hive-site.xml

切换到 /opt/hive/hive2.1/conf 目录下，将hive-default.xml.template 拷贝一份，并重命名为hive-site.xml，然后编辑hive-site.xml文件

```
cp hive-default.xml.template hive-site.xml
vim hive-site.xml
```

添加如下：

```
<!-- 指定HDFS中的hive仓库地址 -->
<property>
  <name>hive.metastore.warehouse.dir</name>
  <value>/root/hive/warehouse</value>
</property>

<property>
  <name>hive.exec.scratchdir</name>
  <value>/root/hive</value>
</property>

<!-- 该属性为空表示嵌入模式或本地模式，否则为远程模式 -->
```

```

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

<!-- 指定mysql的连接 -->
<property>
  <name>javax.jdo.option.ConnectionURL</name>
  <value>jdbc:mysql://master:3306/hive?
createDatabaseIfNotExist=true</value>
</property>
<!-- 指定驱动类 -->
<property>
  <name>javax.jdo.option.ConnectionDriverName</name>
  <value>com.mysql.jdbc.Driver</value>
</property>
<!-- 指定用户名 -->
<property>
  <name>javax.jdo.option.ConnectionUserName</name>
  <value>root</value>
</property>
<!-- 指定密码 -->
<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>123456</value>
</property>
<property>
<name>hive.metastore.schema.verification</name>
<value>false</value>
<description>
</description>
</property>

# 将配置中所有的{system:java.io.tmpdir}更改为/opt/hive/tmp，并将此文件赋予读写权限
# 将{syatem:user:name}更改为root

```

3. 修改hive-env.sh

修改hive-env.sh 文件，没有就复制 hive-env.sh.template ，并重命名为hive-env.sh

```

cp hive-env.sh.template hive-env.sh
vim hive-env.sh

```

在配置文件中添加（根据实际情况填写）：

```

export HADOOP_HOME=/usr/local/hadoop2.7.2
export HIVE_CONF_DIR=/opt/hive/hive2.3/conf
export HIVE_AUX_JARS_PATH=/opt/hive/hive2.3/lib

```

4. 添加数据驱动包

由于Hive 默认自带的数据库是使用mysql，所以这块就是用mysql 将mysql 的驱动包 上传到 /opt/hive/hive2.3/lib

使用ftp将mysql-connector-java-8.0.21.jar传到lib目录下。

Hive Shell测试

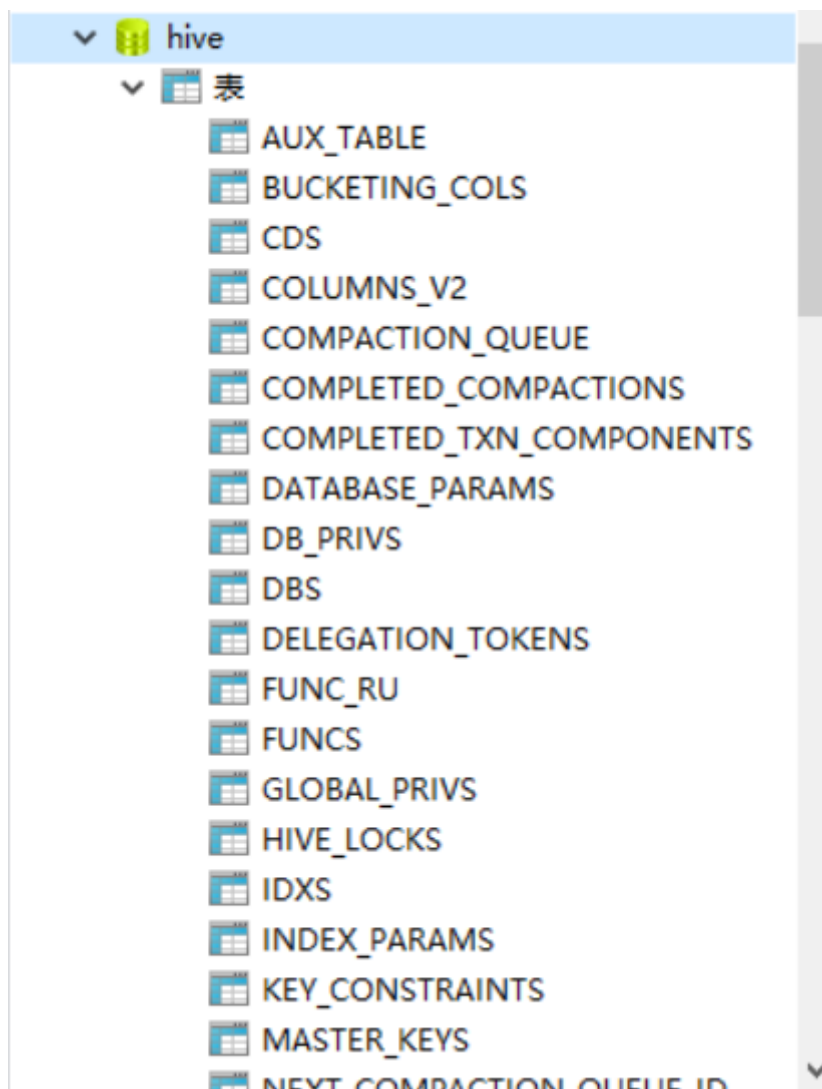
在成功启动Hadoop之后，切换到Hive目录下

```
cd /opt/hive/hive2.1/bin
```

初始化数据库

```
schematool -initSchema -dbType mysql
```

执行成功之后，可以看到hive数据库和一堆表已经创建成功了：



切换到 bin目录

```
cd /opt/hive/hive2.1/bin
```

进入hive

```
hive
```

```
Logging initialized using configuration in jar:file:/usr/local/hive2.3.6/lib/hive-common-2.3.6.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
hive> █
```