

centos7.6搭建hive-2.3.4

1.软件准备

系统：centos7.6

硬件架构：aarch64

服务器：华为泰山2280v2

下载mysql驱动，驱动需要和我们的mysql版本匹配。注意，如果是8的驱动，驱动类需要写成“com.mysql.cj.jdbc.Driver”。驱动是5版本则写成“com.mysql.jdbc.Driver”

下载地址：<http://central.maven.org/maven2/mysql/mysql-connector-java/8.0.13/>

下载hive安装包。下载文件为：apache-hive-2.3.4-bin.tar.gz

下载地址：<https://archive.apache.org/dist/hive/hive-2.3.4/>

mysql安装过程不在此文档

2.安装hive2.3.4

选择hive安装目录为/opt/bigdata

```
[root@master ~]# tar -zxf apache-hive-2.3.4-bin.tar.gz -C /opt/bigdata #解压hive
[root@master ~]# cd /opt/bigdata
[root@master ~]# mv apache-hive-2.3.4-bin hive-2.3.4 #重命名文件夹
```

设置hive的环境变量

```
[root@master ~]# vim /etc/profile
export HIVE_HOME=/opt/bigdata/hive-2.3.4
export PATH=$PATH:$HIVE_HOME/bin

[root@master ~]# source /etc/profile
```

修改hive配置文件

```
[root@master ~]# cd /opt/bigdata/hive-2.3.4/conf
[root@master conf]# mv hive-env.sh.template hive-env.sh #重命名环境文件
[root@master conf]# mv hive-log4j2.properties.template hive-log4j2.properties#重命名日志文件
[root@master conf]# cp hive-default.xml.template hive-site.xml #拷贝生成xml文件
```

修改hive-env.sh文件

将第48行改为 HADOOP_HOME=/opt/bigdata/hadoop-2.8.5 (此路径为hadoop的安装路径)

将第51行改为 export HIVE_CONF_DIR=/opt/bigdata/hive-2.3.4/conf (此路径为hive配置文件路径)

修改hive-log4j2.properties文件

将第24行改为 property.hive.log.dir =/opt/bigdata/hive-2.3.4/logs

修改hive-site.xml中的文件内容,查找name, 然后替换value, 如下:

```
[root@master conf]# vim hive-site.xml
<configuration>
<!-- hive元数据地址, 默认是/user/hive/warehouse -->
<property>
    <name>hive.metastore.warehouse.dir</name>
    <value>/user/hive/warehouse</value>
</property>
<!-- hive查询时输出列名 -->
<property>
    <name>hive.cli.print.header</name>
    <value>true</value>
</property>
<!-- 显示当前数据库名 -->
<property>
    <name>hive.cli.print.current.db</name>
    <value>true</value>
</property>
<!-- 开启本地模式, 默认是false -->
<property>
    <name>hive.exec.mode.local.auto</name>
    <value>true</value>
</property>
<!-- URL用于连接远程元数据 -->
<property>
    <name>hive.metastore.uris</name>
    <value>thrift://deptest37:9083</value>
    <description>Thrift URI for the remote metastore. Used by metastore client
to connect to remote metastore.</description>
</property>
<!-- 元数据使用mysql数据库 你数据库所在的主机-->
<property>
    <name>javax.jdo.option.ConnectionURL</name>
    <value>jdbc:mysql://master:3306/hive?
createDatabaseIfNotExist=true&useSSL=false</value>
    <description>JDBC connect string for a JDBC metastore</description>
</property>
<!-- 元数据使用mysql数据库 数据库用户名-->
<property>
    <name>javax.jdo.option.ConnectionUserName</name>
    <value>root</value>
    <description>username to use against metastore database</description>
</property>
<!-- 元数据使用mysql数据库 数据库登录密码-->
<property>
    <name>javax.jdo.option.ConnectionPassword</name>
    <value>123456</value>
    <description>password to use against metastore database</description>
</property>
<property>
    <name>javax.jdo.option.ConnectionDriverName</name>
    <value>com.mysql.cj.jdbc.Driver</value>
    <description>Driver class name for a JDBC metastore</description>
</property>
</configuration>
```

创建hive元数据所在路径，前提是已经安装好了hadoop，否则无法创建。

```
[root@master ~]# hdfs dfs -mkdir /tmp #如果有这个路径，这不需要重新创建
[root@master ~]# hdfs dfs -mkdir -p /user/hive/warehouse #创建目录
[root@master ~]# hdfs dfs -chmod g+w /tmp #修改文件权限
[root@master ~]# hdfs dfs -chmod g+w /user/hive/warehouse #修改文件权限
```

拷贝mysql驱动到hive的lib中

```
[root@master ~]# cp mysql-connector-java-8.0.13.jar $HOME_HIVE/lib
```

初始化元数据库

```
[root@master ~]# schematool -initSchema -dbType mysql //初始化成功最后会出现如下内容
Starting metastore schema initialization to 2.3.0
Initialization script hive-schema-2.3.0.mysql.sql
Initialization script completed
schemaTool completed
```

开启元数据

```
[root@master ~]# nohup hive --service metastore >> ~/metastore.log 2>&1 &
```

开启hive

```
[root@master ~]# nohup hive --service hiveserver2 >> ~/hiveserver2.log 2>&1 &
```

测试hive

```
[root@master ~]# hive
which: no hbase in (/usr/local/lib/java/jdk1.8.0_221/bin:/usr/lib64/qt-3.3/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/opt/bigdata/hadoop-2.8.5/sbin:/opt/bigdata/hadoop-2.8.5/bin:/opt/bigdata/zookeeper-3.4.13/bin:/opt/bigdata/hive-2.3.4/bin:/usr/local/mysql/bin:/opt/bigdata/hive-2.3.4/bin:/opt/bigdata/maven/bin:/opt/bigdata/hadoop-2.8.5/bin:/root/bin)
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/bigdata/hive-2.3.4/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/bigdata/hadoop-2.8.5/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Logging initialized using configuration in file:/opt/bigdata/hive-2.3.4/conf/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 (default)> show databases;
OK
database_name
default
xgc
```

```
Time taken: 1.312 seconds, Fetched: 2 row(s)  
hive (default)>
```