

在Ubuntu上安装hive

【注意】以下所有操作没有特别强调的都是在A节点上

1.安装MySQL

(1) 在线安装MySQL

```
1 $sudo apt-get install mysql-server
2 $sudo apt-get install mysql-client
3 $sudo apt-get install libmysqlclient-dev
```

(2) 连接MySQL数据库

```
1 $ sudo mysql -uroot -p
```

```
1 alter user 'root'@'localhost' identified by 'root';
2 UPDATE mysql.user SET plugin='mysql_native_password' WHERE user='root' and host =
'localhost';
3 FLUSH PRIVILEGES ;
4 exit;
```

```
1 $ mysql -uroot -proot
```

(3) 创建用户

```
1 CREATE USER 'hive'@'%' IDENTIFIED BY 'hive';
2 CREATE DATABASE hive;
3 GRANT ALL PRIVILEGES ON hive.* TO 'hive'@'%' ;
4 FLUSH PRIVILEGES ;
5 exit
```

(4) 修改MySQL配置文件，允许远程访问

```
1 $ sudo gedit /etc/mysql/mysql.conf.d/mysqld.cnf
```

将bind-address = 127.0.0.1行注释掉

(5) 重启MySQL服务

```
1 $sudo service mysql restart
```

2.安装Hive

(1) 软件下载

(2) 解压Hive安装文件

```
1 $ sudo tar -zxvf apache-hive-3.1.2-bin.tar.gz -C /usr/local
2 $ cd /usr/local
3 $ sudo mv /usr/local/apache-hive-3.1.2-bin/ ./hive
4 $ sudo chown -R hadoop:hadoop ./hive
```

(3) 配置环境变量，并使其生效

```
1 $gedit ~/.bashrc
```

写入下列信息：

```
1 export HIVE_HOME=/usr/local/hive
2 export PATH=$PATH:$HIVE_HOME/bin
```

```
1 $ source ~/.bashrc
```

(4) 添加MySQL驱动

```
1 $ mv mysql-connector-java-8.0.33.jar /usr/local/hive/lib/
```

(5) 创建、配置hive-site.xml文件

```
1 $ cd /usr/local/hive/conf
2 $ mv hive-default.xml.template ./hive-site.xml
3 $ gedit ./hive-site.xml
```

在文件末尾的</configuration>标签之前添加下列内容

```
1 <property>
2     <name>hive.metastore.warehouse.dir</name>
3     <value>/user/hive/warehouse</value>
4 </property>
5 <property>
6     <name>fs.defaultFS</name>
7     <value>hdfs://mashuai01:8020</value><!--这里是HDFS的访问地址-->
8 </property>
9 <property>
10    <name>javax.jdo.option.ConnectionURL</name>
11    <value>jdbc:mysql://localhost:3306/hive?
12    createDatabaseIfNotExist=true&useUnicode=true&characterEncoding=UTF-8</value>
13 </property>
14 <property>
15    <name>javax.jdo.option.ConnectionDriverName</name>
16    <value>com.mysql.jdbc.Driver</value>
17 </property>
18 <property>
19    <name>javax.jdo.option.ConnectionPassword</name>
20    <value>hive</value>
21 </property>
22 <property>
23    <name>javax.jdo.option.ConnectionUserName</name>
```

```
23     <value>hive</value>
24 </property>
```

(6) 修改hive-site.xml文件中的配置项

将hive-site.xml文件中含有“system:java.io.tmpdir”的配置项的值修改为/usr/local/hive/tmp
有如下4处

(7) 创建、配置hive-env.sh 文件

```
1 $ cd /usr/local/hive/conf
2 $ mv hive-env.sh.template  hive-env.sh
3 $ gedit  hive-env.sh
```

在hive-env.sh文件中加入下列配置信息：

```
1 export JAVA_HOME=/usr/local/jdk1.8.0_181
2 export HIVE_HOME=/usr/local/hive
3 export HADOOP_HOME=/usr/local/hadoop
```

(8) [A,B,C]修改Hadoop集群配置core-site.xml，并重启集群

```
1 <property>
2     <name>hadoop.proxyuser.hadoop.groups</name>
3     <value>*</value>
4 </property>
5 <property>
6     <name>hadoop.proxyuser.hadoop.hosts</name>
7     <value>*</value>
8 </property>
9
```

```
1 $ stop-all.sh
2 $ start-all.sh
```

(9) Hive初始化

```
1 $ schematool -dbType mysql -initSchema
```

(10) 启动远程服务HiveServer 2

```
1 $ hiveserver2 &
```

(11) 打开控制台命令行工具

```
1 $ hive
```

或者

```
1 beeline -u jdbc:hive2://mashuai01:10000 -n hadoop -p hadoop
```

注意mashuai01改为你自己的主机名

连接hive后，效果如下图。

```
hadoop@master:~$ beeline -u jdbc:hive2://master:10000/ -n hadoop -p hadoop
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.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]
Connecting to jdbc:hive2://master:10000/
Connected to: Apache Hive (version 3.1.2)
Driver: Hive JDBC (version 3.1.2)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 3.1.2 by Apache Hive
0: jdbc:hive2://master:10000/>
```

附录

1.如果出现如下错误NoSuchMethodError

Exception in thread "main" java.lang.NoSuchMethodError:

com.google.common.base.Preconditions.checkArgument(ZLjava/lang/String;Ljava/lang/Object;)V

#解决办法: guava-19.0.jar和hadoopjar包冲突(用hadoop的guava-27.0-jre.jar替换掉hive下的guava-19.0.jar包)

#先找到对应的jar包然后替换

find / -name guava*

用hadoop的guava-27.0-jre.jar替换掉hive下的guava-19.0.jar包

2.使用vim时如何定位指定的行

vim 进入指定的行 :+行号, 或者 vim +行号 文件名

3.建表demo

```
1 create table inner_emp(  
2     empno int,  
3     nname char(20),  
4     job string,  
5     mgr int,  
6     hiredate string,  
7     sal float,  
8     comm float,  
9     deptno int)  
10 ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;
```

```
1 create table inner_dept(  
2     deptno int,  
3     dname string,  
4     loc string)
```

```
2 deptno int,  
3 dname string,  
4 manager_id int)  
5 ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;
```

```
1 create external table ext_emp(  
2     empno int,  
3     ename char(20),  
4     job string,  
5     mgr int,  
6     hiredate string,  
  
7     sal float,  
8     comm float,  
9     deptno int)  
10 ROW FORMAT DELIMITED FIELDS TERMINATED BY ','  
  
11 LOCATION '/user/hive/warehouse/ext_data';
```