hadoop2.2+hbase0.96+hive0.12

安装整合详细高可靠文档及经验总结

本文链接：http://www.aboutyun.com/thread-7890-1-1.html

问题导读：

1.hadoop 的安装需要安装哪些如那件？

2.hadoop 与 hbase 整合需要注意哪些问题？

3.hive 与 hbase 的通信包是哪个？

网上这个文档非常流行，但是感觉缺乏一些细节的指导，除非对 Linux 和 hadoop 有一定的认识。否则新手很难从这些文档中安装成功。这里重新实践了一下，hadoop2.2 ,hbase0.96,hive0.12 的安装及他们的整合，这里整理一下。

1.首先 hadoop2.2 的安装

hadoop2.2 完全分布式最新高可靠安装文档

http://www.aboutyun.com/thread-7684-1-1.html

这个文档里面包含了很多的知识：

包括 ssh 无密码互通，jdk 的安装，以及 shell 脚本的编写，当然还有文档的配置。相信新手可以通过这个文档安装成功。

2.hbase 0.96 整合到 hadoop2.2

hbase 0.96 整合到 hadoop2.2 三个节点全分布式安装高可靠文档

http://www.aboutyun.com/thread-7746-1-1.html

这个的整合需要注意权限及包的整合，一些临时目录，最好都建立好，而不是让系统建立，同时建立的时候不要放到系统默认文件夹，这个可能重启就会产生问题，导致缺少 hmaster 或则其他进程。还有个问题，有的同学安装完毕之后，hmaster 或则 regionserver 起来之后接着宕机，是因为文件夹的权限问题，最好临时目录等都放到 hbase 的安装目录中。
同时还需要注意 hbase 与 hadoop 版本要一致，否则会出问题。下面帖子可以参考：
hadoop2.2.0+hbase0.94.18 出现问题

3.hbase 0.96 与 hive0.12 整合
hbase0.96 与 hive0.12 整合高可靠文档及问题总结
http://www.aboutyun.com/thread-7881-1-1.html

这个的整合遇到不少问题，与其 hadoop,hbase 不一样的地方是它有一个客户端和服务器端，这样导致很多新手问这个问题，到底是否需要每台电脑都需要安装 mysql。这个是不需要的，mysql 只需要安装在服务器端就可以了。这两个整合涉及到了两个包，还是比较重要的，包括：
mysql 的连接包 mysql-connector-java-5.1.10-bin.jar，
还有就是 hbase 与 hive 的通信包
hive-hbase-handler-0.13.0.jar
这两个包都需要复制到 hive 安装目录的 lib 文件夹下，并且注意他们之间版本要匹配。

这里面对他们的整合做一个简单的描述和总结，更详细的内容，可以查看文档。

对于 **hadoop2.2 的安装可参考：**
hadoop2.2 完全分布式最新高可靠安装文档
http://www.aboutyun.com/thread-7684-1-1.html
hbase 0.96 整合到 hadoop2.2 可参考：
hbase 0.96 整合到 hadoop2.2 三个节点全分布式安装高可靠文档
http://www.aboutyun.com/thread-7746-1-1.html

下文详细讲解了 hbase0.96 及 hive0.12 的整合文档：
链接地址为：http://www.aboutyun.com/thread-7881-1-1.html

hbase0.96 与 hive0.12 整合高可靠文档及问题总结

问题导读：

1.hive 安装是否需要安装 mysql？
2.hive 是否分为客户端和服务器端？
3.hive 的元数据库有哪两种？
4.hive 与 hbase 整合的关键是什么？
5.hive 的安装是否必须安装 hadoop?
6.hive 与 hbase 整合需要做哪些准备工作？

网上有很多资料，看到大部分都是一致的，看到一篇国外的文章，原来都是翻译的，并没有经过实践。这里记录一下实践的过程。
本篇是在：
hadoop2.2 完全分布式最新高可靠安装文档
hbase 0.96 整合到 hadoop2.2 三个节点全分布式安装高可靠文档

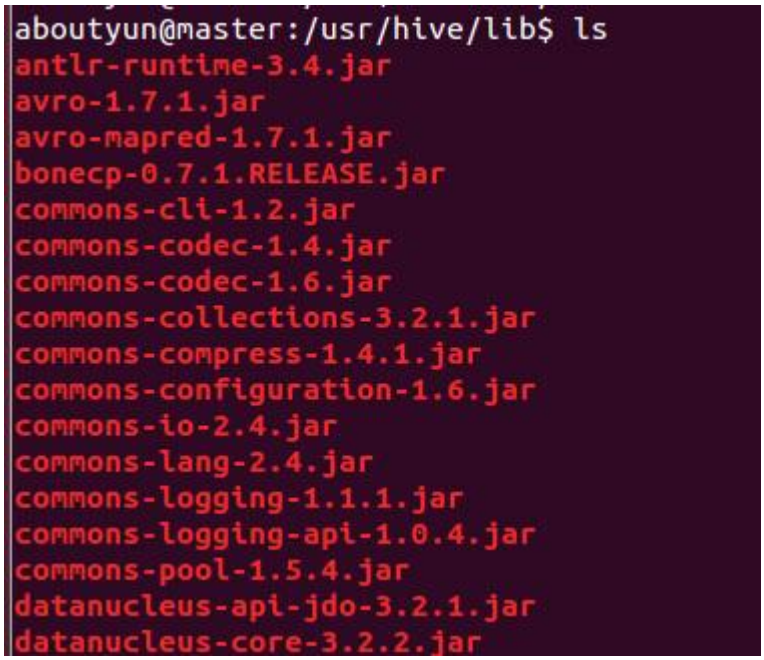
基础上的一个继续：
因为 derby 数据库使用的局限性，我们采用 mysql 作为元数据库。
derby 存在什么缺陷
1.derby 不能多个客户端登录
2.derby 登录必须在相同目录下，否则可能会找不到所创建的表。
比如在/hive 目录下启动 hive 程序，那么所创建的表就会存储在/hive 下面保存。如果在/home 下面，所创建的表就会在/home 下面保存。这样导致初学者摸不着头脑。如果还不是不太明白可以，可以参考
---
hive 使用 derby 作为元数据库找达到所创建表的原因
http://www.aboutyun.com/thread-7803-1-1.html

下面我们开始安装：
1.下载 hivehive
链接: http://pan.baidu.com/s/1eQw0o50 密码: mgy6
2. 安装：
tar zxvf hive-0.12.0.tar.gz
重命令名为：hive 文件夹
达到如下效果：

3. 替换 jar 包，与 hbase0.96 和 hadoop2.2 版本一致

由于我们下载的 hive 是基于 hadoop1.3 和 hbase0.94 的，所以必须进行替换，因为我们的 hbse0.96 是基于 hadoop2.2 的，所以我们必须先解决 hive 的 hadoop 版本问题，目前我们从官网下载的 hive 都是用 1.几的版本编译的，因此我们需要自己下载源码来用 hadoop2.X 的版本重新编译 hive，这个过程也很简单，只需要如下步骤：

（1）进入/usr/hive/lib



上面只是截取了一部分：

（2）同步 hbase 的版本

先 cd 到 hive0.12.0/lib 下，将 hive-0.12.0/lib 下 hbase-0.94 开头的那两个 jar 包删掉，然后从/home/hadoop/hbase-0.96.0-hadoop2/lib 下 hbase 开头的包都拷贝过来

find /usr/hbase/hbas/lib -name "hbase*.jar"|xargs -i cp {} ./

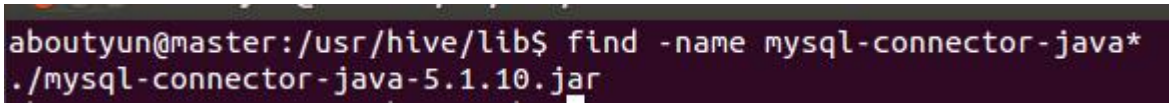
（3）基本的同步完成了

重点检查下 zookeeper 和 protobuf 的 jar 包是否和 hbase 保持一致，如果不一致，
拷贝 protobuf.**.jar 和 zookeeper-3.4.5.jar 到 hive/lib 下。

（4）用 mysql 当原数据库，

找一个 mysql 的 jdbcjar 包 mysql-connector-java-5.1.10-bin.jar 也拷贝到 hive-0.12.0/lib 下

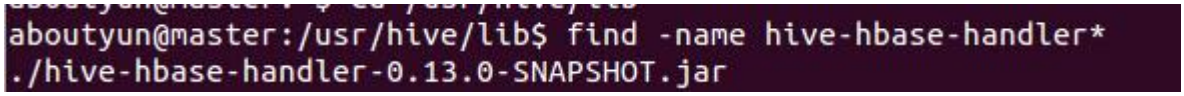
可以通过下面命令来查找是否存在



如果不存在则下载：
链接: http://pan.baidu.com/s/1gdCDoGj 密码: 80yl

注意 mysql-connector-java-5.1.10-bin.jar
修改权限为 777 （chmod 777 mysql-connector-java-5.1.10-bin.jar）

还有，看一下 hbase 与 hive 的通信包是否存在：



可以通过下面命令：

aboutyun@master:/usr/hive/lib$ find -name hive-hbase-handler*
./hive-hbase-handler-0.13.0-SNAPSHOT.jar
不存在则下载：

链接: http://pan.baidu.com/s/1gd9p0Fh 密码: 94g1

4. 安装 mysql

- Ubuntu 采用 apt-get 安装
- sudo apt-get install mysql-server
- 建立数据库 hive
- create database hivemeta
- 创建 hive 用户,并授权
- grant all on hive.* to hive@'%' identified by 'hive';
- flush privileges;

对于 musql 的安装不熟悉，可以参考：

Ubuntu 下面卸载以及安装 mysql
http://www.aboutyun.com/thread-7788-1-1.html

上面命令解释一下：
• sudo apt-get install mysql-server 安装数据服务器，如果想尝试通过其他客户端远程连接，则还需要安装 mysql-client

• create database hivemeta
这个使用来存储 hive 元数据，所创建的数据库

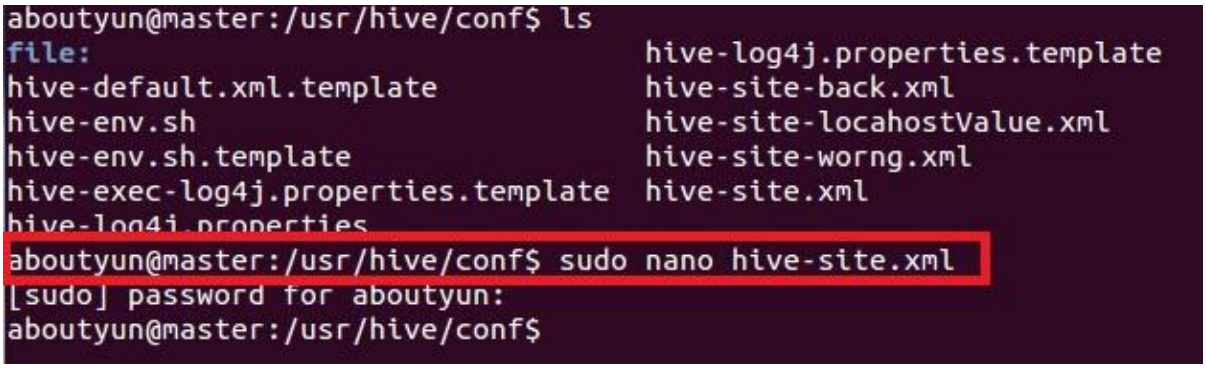
• grant all on hive.* to hive@'%' identified by 'hive'; 这个是授权，还是比较重要的，否则 hive 客户端远程连接会失败
里面的内容不要照抄：需要根据自己的情况来修改。上面的用户名和密码都为 hive。

如果连接不成功尝试使用 root 用户

```
1.  grant all on hive.* to 'root'@'%'identified by '123';
2.  flush privileges;
```

---

4. 修改 hive-site 文件配置：



下面配置需要注意的是：
（1）使用的是 mysql 的 root 用户，密码为 123，如果你是用的 hive，把用户名和密码该为 hive 即可：

```
<property>
  <name>javax.jdo.option.ConnectionUserName</name>
  <value>root</value>
</property>
<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>123</value>
</property>
```

（2）hdfs 新建文件并授予权限

```
<property>
  <name>hive.metastore.warehouse.dir</name>
  <value>hdfs://master:8020/hive/warehouse</value>
</property>
<property>
  <name>hive.exec.scratchdir</name>
  <value>hdfs://master:8020/hive/scratchdir</value>
</property>
```

对于上面注意

bin/hadoop fs -mkdir /hive/warehouse
bin/hadoop fs -mkdir /hive/scratchdir
bin/hadoop fs -chmod g+w /hive/warehouse
bin/hadoop fs -chmod g+w /hive/scratchdir

（3）hive.aux.jars.path 切忌配置正确

不能有换行或则空格。特别是换行，看到很多文章都把他们给分开了，这对很多新手是一个很容易掉进去的陷阱。

```
1. <property>
2.   <name>hive.aux.jars.path</name>
3.   <value>file:///usr/hive/lib/hive-hbase-handler-0.13.0-SNAPSHOT.jar,fi
   le:///usr/hive/lib/protobuf-java-2.5.0.jar,file:///usr/hive/lib/hbase-c
   lient-0.96.0-hadoop2.jar,file:///usr/hive/lib/hbase-common-0.96.0-hadoo
   p2.jar,file:///usr/hive/lib/zookeeper-3.4.5.jar,file:///usr/hive/lib/gu
   ava-11.0.2.jar</value>
4. </property>
```

```
<property>
  <name>hive.aux.jars.path</name>
  <value>file:///usr/hive/lib/hive-hbase-handler-0.13.0-SNAPSHOT.jar,file:///usr/
hive/lib/protobuf-java-2.5.0.jar,file:///usr/hive/lib/hbase-client-0.96.0-hadoop2
.jar,file:///usr/hive/lib/hbase-common-0.96.0-hadoop2.jar,file:///usr/hive/lib/zo
okeeper-3.4.5.jar,file:///usr/hive/lib/guava-11.0.2.jar</value>
</property>
```

上面问题解决，把下面内容放到 **hive-site** 文件即可

这里介绍两种配置方式，一种是远程配置，一种是本地配置。最好选择远程配置

远程配置

```
1.  <configuration>
2.  <property>
3.    <name>hive.metastore.warehouse.dir</name>
4.    <value>hdfs://master:8020/hive/warehouse</value>
5.  </property>
6.  <property>
7.    <name>hive.exec.scratchdir</name>
8.    <value>hdfs://master:8020/hive/scratchdir</value>
9.  </property>
10. <property>
11.   <name>hive.querylog.location</name>
12.   <value>/usr/hive/logs</value>
13. </property>
14. <property>
15.   <name>javax.jdo.option.ConnectionURL</name>
16.   <value>jdbc:mysql://172.16.77.15:3306/hiveMeta?createDatabaseIfNotExi
    st=true</value>
17. </property>
18. <property>
```

```
19.   <name>javax.jdo.option.ConnectionDriverName</name>
20.   <value>com.mysql.jdbc.Driver</value>
21. </property>
22. <property>
23.   <name>javax.jdo.option.ConnectionUserName</name>
24.   <value>hive</value>
25. </property>
26. <property>
27.   <name>javax.jdo.option.ConnectionPassword</name>
28.   <value>hive</value>
29. </property>
30. <property>
31.   <name>hive.aux.jars.path</name>
32.   <value>file:///usr/hive/lib/hive-hbase-handler-0.13.0-SNAPSHOT.jar,fi
    le:///usr/hive/lib/protobuf-java-2.5.0.jar,file:///usr/hive/lib/hbase-c
    lient-0.96.0-hadoop2.jar,file:///usr/hive/lib/hbase-common-0.96.0-hadoo
    p2.jar,file:///usr/hive/lib/zookeeper-3.4.5.jar,file:///usr/hive/lib/gu
    ava-11.0.2.jar</value>
33. </property>
34. <property>
35.   <name>hive.metastore.uris</name>
36.   <value>thrift://172.16.77.15:9083</value>
37. </property>
38. </configuration>
```

本地配置：

```
1.  <configuration>
2.  <property>
3.    <name>hive.metastore.warehouse.dir</name>
4.    <value>/user/hive_remote/warehouse</value>
5.  </property>
```

```
6.
7.  <property>
8.    <name>hive.metastore.local</name>
9.    <value>true</value>
10. </property>
11.
12. <property>
13.   <name>javax.jdo.option.ConnectionURL</name>
14.   <value>jdbc:mysql://localhost/hive_remote?createDatabaseIfNotExist=tr
    ue</value>
15. </property>
16.
17. <property>
18.   <name>javax.jdo.option.ConnectionDriverName</name>
19.   <value>com.mysql.jdbc.Driver</value>
20. </property>
21.
22. <property>
23.   <name>javax.jdo.option.ConnectionUserName</name>
24.   <value>root</value>
25. </property>
26.
27. <property>
28.   <name>javax.jdo.option.ConnectionPassword</name>
29.   <value>123</value>
30. </property>
31. </configuration>
```

---

5. 修改其它配置：

1.修改 hadoop 的 hadoop-env.sh(否则启动 hive 汇报找不到类的错误)

```
# Set HADOOP_HOME to point to a specific hadoop install directory
export  HADOOP_HOME=/usr/hadoop

# Hive Configuration Directory can be controlled by:
export HIVE_CONF_DIR=/usr/hive/conf
```

2.修改$HIVE_HOME/bin 的 hive-config.sh，增加以下三行

```
# processes --config option from command line
#
  export JAVA_HOME=/usr/jdk1.7
  export HIVE_HOME=/usr/hive
  export HADOOP_HOME=/usr/hadoop
```

****

首先说一些遇到的各种问题

1.遇到的问题

问题 1：元数据库未启动

这里首先概括一下，会遇到的问题。首先需要启动元数据库，通过下面命令：

（1）hive --service metastore

（2）hive --service metastore -hiveconf hive.root.logger=DEBUG,console

注释：

-hiveconf hive.root.logger=DEBUG,console 命令的含义是进入 debug 模式，便于寻找错误

如果不启用元数据库，而是使用下面命令

```
1.  hive
```

你会遇到下面错误

```
1.  Exception in thread "main" java.lang.RuntimeException:
    java.lang.RuntimeException: Unable to instantiate
    org.apache.hadoop.hive.metastore.HiveMetaStoreClient
2.          at
    org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:
    295)
3.          at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:679)
4.          at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:623)
5.          at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
6.          at
    sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.ja
    va:57)
7.          at
    sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccesso
    rImpl.java:43)
8.          at java.lang.reflect.Method.invoke(Method.java:606)
9.          at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
10. Caused by: java.lang.RuntimeException: Unable to instantiate
    org.apache.hadoop.hive.metastore.HiveMetaStoreClient
11.         at
    org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUt
    ils.java:1345)
12.         at
    org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(Retryin
    gMetaStoreClient.java:62)
13.         at
    org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(Retry
    ingMetaStoreClient.java:72)
```

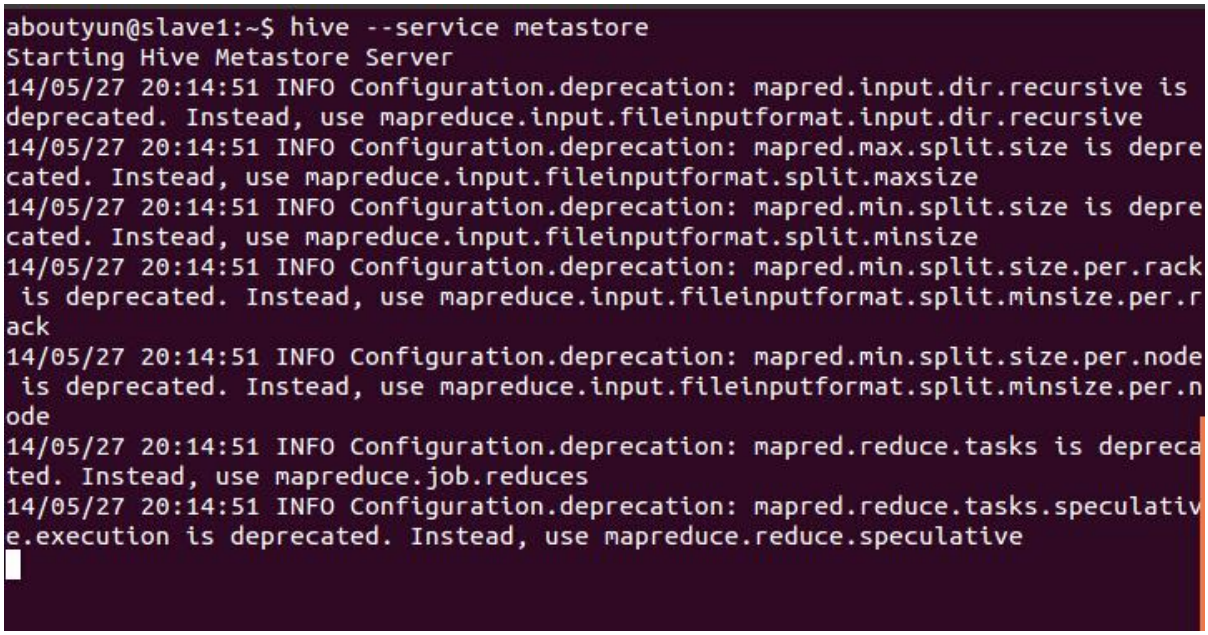
```
14.         at
    org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java
    :2420)
15.         at
    org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:2432)
16.         at
    org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:
    289)
17.         ... 7 more
18. Caused by: java.lang.reflect.InvocationTargetException
19.         at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native
    Method)
20.         at
    sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructor
    AccessorImpl.java:57)
21.         at
    sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingCon
    structorAccessorImpl.java:45)
22.         at
    java.lang.reflect.Constructor.newInstance(Constructor.java:526)
23.         at
    org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUt
    ils.java:1343)
24.         ... 12 more
25. Caused by: MetaException(message:Could not connect to meta store using any
    of the URIs provided. Most recent failure:
    org.apache.thrift.transport.TTransportException:
26.
27.
28. java.net.ConnectException: Connection refused
29.         at org.apache.thrift.transport.TSocket.open(TSocket.java:185)
```

```
30.         at
    org.apache.hadoop.hive.metastore.HiveMetaStoreClient.open(HiveMetaStore
    Client.java:288)
31.         at
    org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaSto
    reClient.java:169)
32.         at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native
    Method)
33.         at
    sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructor
    AccessorImpl.java:57)
34.         at
    sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingCon
    structorAccessorImpl.java:45)
35.         at
    java.lang.reflect.Constructor.newInstance(Constructor.java:526)
36.         at
    org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUt
    ils.java:1343)
37.         at
    org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(Retryin
    gMetaStoreClient.java:62)
38.         at
    org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(Retry
    ingMetaStoreClient.java:72)
39.         at
    org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java
    :2420)
40.         at
    org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:2432)
```

```
41.         at
    org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:
    289)
42.         at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:679)
43.         at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:623)
44.         at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
45.         at
    sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.ja
    va:57)
46.         at
    sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccesso
    rImpl.java:43)
47.         at java.lang.reflect.Method.invoke(Method.java:606)
48.         at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
49. Caused by: java.net.ConnectException: Connection refused
50.         at java.net.PlainSocketImpl.socketConnect(Native Method)
51.         at
    java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java
    :339)
52.         at
    java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketIm
    pl.java:200)
53.         at
    java.net.AbstractPlainSocketImpl.connect(AbstractPlainSocketImpl.java:1
    82)
54.         at java.net.SocksSocketImpl.connect(SocksSocketImpl.java:392)
55.         at java.net.Socket.connect(Socket.java:579)
56.         at org.apache.thrift.transport.TSocket.open(TSocket.java:180)
57.         ... 19 more
58. )
```

```
59.         at
    org.apache.hadoop.hive.metastore.HiveMetaStoreClient.open(HiveMetaStore
    Client.java:334)
60.         at
    org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaSto
    reClient.java:169)
61.         ... 17 more
```

问题 2：元数据库启动状态是什么样子的




```
1.  hive --service metastore
2.  Starting Hive Metastore Server
3.  14/05/27 20:14:51 INFO Configuration.deprecation:
    mapred.input.dir.recursive is deprecated. Instead, use
    mapreduce.input.fileinputformat.input.dir.recursive
4.  14/05/27 20:14:51 INFO Configuration.deprecation: mapred.max.split.size is
    deprecated. Instead, use mapreduce.input.fileinputformat.split.maxsize
5.  14/05/27 20:14:51 INFO Configuration.deprecation: mapred.min.split.size is
    deprecated. Instead, use mapreduce.input.fileinputformat.split.minsize
```

```
6.  14/05/27 20:14:51 INFO Configuration.deprecation:
    mapred.min.split.size.per.rack is deprecated. Instead, use
    mapreduce.input.fileinputformat.split.minsize.per.rack
7.  14/05/27 20:14:51 INFO Configuration.deprecation:
    mapred.min.split.size.per.node is deprecated. Instead, use
    mapreduce.input.fileinputformat.split.minsize.per.node
8.  14/05/27 20:14:51 INFO Configuration.deprecation: mapred.reduce.tasks is
    deprecated. Instead, use mapreduce.job.reduces
9.  14/05/27 20:14:51 INFO Configuration.deprecation:
    mapred.reduce.tasks.speculative.execution is deprecated. Instead, use
    mapreduce.reduce.speculative
```

刚开始遇到这种情况，我知道是因为可能没有配置正确，这个耗费了很长时间，一直没有找到正确的解决方案。当再次执行

hive --service metastore

命令的时候报 4083 端口被暂用： 报错如下红字部分。表示 9083 端口已经被暂用，也就是说客户端已经和主机进行了通信，当我在进行输入 hive 命令的时候，进入下面图 1 界面

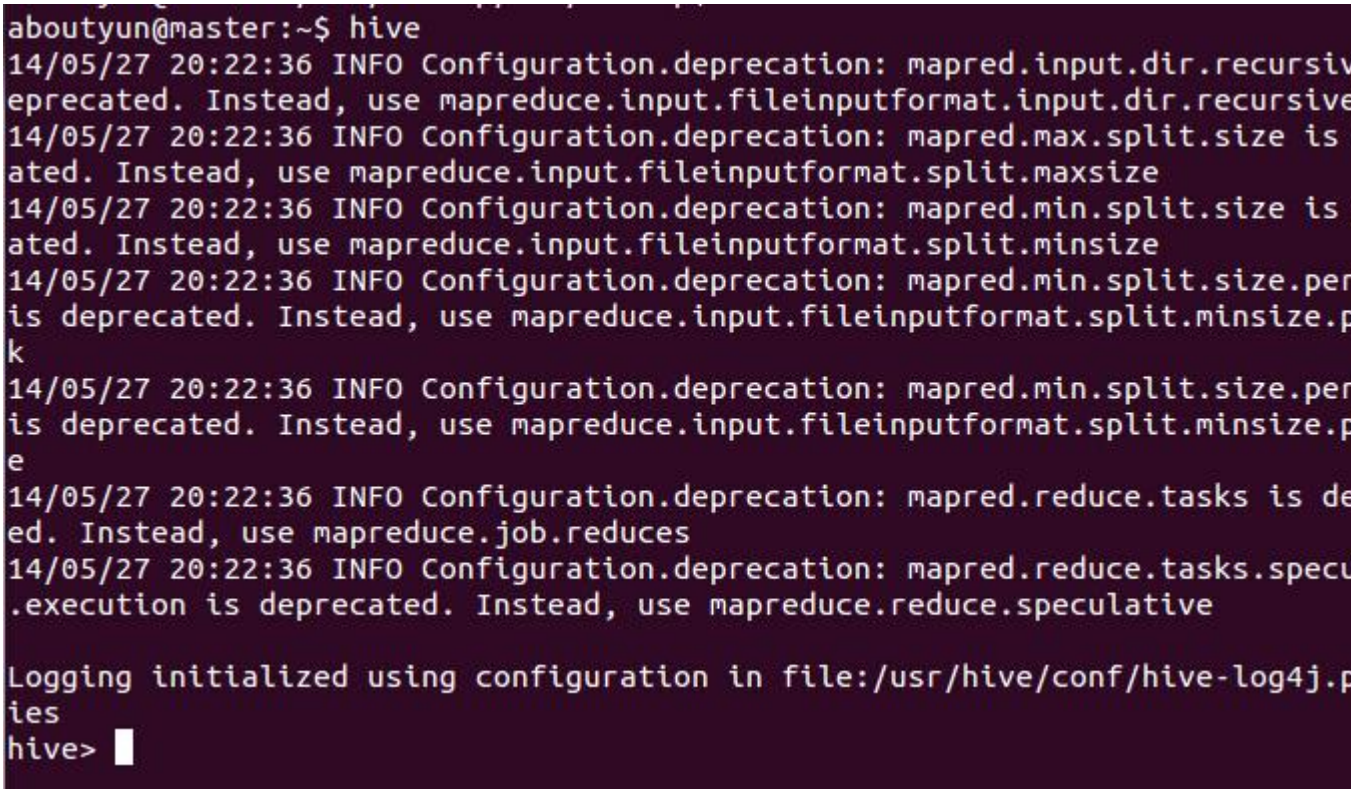


图 1

```
1.  Could not create ServerSocket on address 0.0.0.0/0.0.0.0:9083.
2.          at
    org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:93)
3.          at
    org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:75)
4.          at
    org.apache.hadoop.hive.metastore.TServerSocketKeepAlive.<init>(TServerS
    ocketKeepAlive.java:34)
5.          at
    org.apache.hadoop.hive.metastore.HiveMetaStore.startMetaStore(HiveMetaS
    tore.java:4291)
6.          at
    org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:
    4248)
7.          at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
8.          at
    sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.ja
    va:57)
9.          at
    sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccesso
    rImpl.java:43)
10.         at java.lang.reflect.Method.invoke(Method.java:606)
11.         at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
12. Exception in thread "main"
    org.apache.thrift.transport.TTransportException: Could not create
    ServerSocket on address 0.0.0.0/0.0.0.0:9083.
13.         at
    org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:93)
14.         at
    org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:75)
```

```
15.         at
    org.apache.hadoop.hive.metastore.TServerSocketKeepAlive.<init>(TServerS
    ocketKeepAlive.java:34)
16.         at
    org.apache.hadoop.hive.metastore.HiveMetaStore.startMetaStore(HiveMetaS
    tore.java:4291)
17.         at
    org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:
    4248)
18.         at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
19.         at
    sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.ja
    va:57)
20.         at
    sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccesso
    rImpl.java:43)
21.         at java.lang.reflect.Method.invoke(Method.java:606)
22.         at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
```

对于端口的暂用，可以采用下面命令杀掉进程

```
1.  netstat -ap|grep 4083
```

上面主要的作用是查出暂用端口的进程 id，然后使用下面命令杀掉进程即可

```
1.  kill -9 进程号
```

详细可以查看下面内容：

使用配置 hadoop 中常用的 Linux 命令

问题 3：hive.aux.jars.path 配置中含有看换行或则空格，报错如下

错误表现 1：/usr/hive/lib/hbase-client-0.96.0-
hadoop2.jar

整个路径错位，导致系统不能识别，这个错位，其实就是换行。

```
1.
```

```
2.  FAILED: Execution Error, return code 1 from
    org.apache.hadoop.hive.ql.exec.mr.MapRedTask
3.
4.
5.  java.io.FileNotFoundException: File does not exist:
    hdfs://hydra0001/opt/module/hive-0.10.0-cdh4.3.0/lib/hive-builtins-0.10
    .0-cdh4.3.0.jar
6.  2014-05-24 19:32:06,563 ERROR exec.Task
    (SessionState.java:printError(440)) - Job Submission failed with exception
    'java.io.FileNotFoundException(File
    file:/usr/hive/lib/hbase-client-0.96.0-
7.  hadoop2.jar does not exist)'
8.  java.io.FileNotFoundException: File
    file:/usr/hive/lib/hbase-client-0.96.0-
9.  hadoop2.jar does not exist
10.         at
    org.apache.hadoop.fs.RawLocalFileSystem.getFileStatus(RawLocalFileSyste
    m.java:520)
11.         at
    org.apache.hadoop.fs.FilterFileSystem.getFileStatus(FilterFileSystem.ja
    va:398)
12.         at org.apache.hadoop.fs.FileUtil.copy(FileUtil.java:337)
13.         at org.apache.hadoop.fs.FileUtil.copy(FileUtil.java:289)
14.         at
    org.apache.hadoop.mapreduce.JobSubmitter.copyRemoteFiles(JobSubmitter.j
    ava:139)
15.         at
    org.apache.hadoop.mapreduce.JobSubmitter.copyAndConfigureFiles(JobSubmi
    tter.java:212)
16.         at
    org.apache.hadoop.mapreduce.JobSubmitter.copyAndConfigureFiles(JobSubmi
    tter.java:300)
```

```
17.         at
    org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter
    .java:387)
18.         at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1268)
19.         at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1265)
20.         at java.security.AccessController.doPrivileged(Native Method)
21.         at javax.security.auth.Subject.doAs(Subject.java:415)
22.         at
    org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformati
    on.java:1491)
23.         at org.apache.hadoop.mapreduce.Job.submit(Job.java:1265)
24.         at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:562)
25.         at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:557)
26.         at java.security.AccessController.doPrivileged(Native Method)
27.         at javax.security.auth.Subject.doAs(Subject.java:415)
28.         at
    org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformati
    on.java:1491)
29.         at
    org.apache.hadoop.mapred.JobClient.submitJobInternal(JobClient.java:557
    )
30.         at
    org.apache.hadoop.mapred.JobClient.submitJob(JobClient.java:548)
31.         at
    org.apache.hadoop.hive.ql.exec.mr.ExecDriver.execute(ExecDriver.java:42
    4)
32.         at
    org.apache.hadoop.hive.ql.exec.mr.MapRedTask.execute(MapRedTask.java:13
    6)
33.         at
    org.apache.hadoop.hive.ql.exec.Task.executeTask(Task.java:152)
```

```
34.         at
    org.apache.hadoop.hive.ql.exec.TaskRunner.runSequential(TaskRunner.java
    :65)
35.         at org.apache.hadoop.hive.ql.Driver.launchTask(Driver.java:1481)
36.         at org.apache.hadoop.hive.ql.Driver.execute(Driver.java:1258)
37.         at
    org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1092)
38.         at org.apache.hadoop.hive.ql.Driver.run(Driver.java:932)
39.         at org.apache.hadoop.hive.ql.Driver.run(Driver.java:922)
40.         at
    org.apache.hadoop.hive.cli.CliDriver.processLocalCmd(CliDriver.java:268
    )
41.         at
    org.apache.hadoop.hive.cli.CliDriver.processCmd(CliDriver.java:220)
42.         at
    org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:422)
43.         at
    org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:790)
44.         at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:684)
45.         at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:623)
46.         at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
47.         at
    sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.ja
    va:57)
48.         at
    sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccesso
    rImpl.java:43)
49.         at java.lang.reflect.Method.invoke(Method.java:606)
50.         at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
51.
52.
```

```
53. 2014-05-24 19:32:06,571 ERROR ql.Driver
    (SessionState.java:printError(440)) - FAILED: Execution Error, return code
    1 from org.apache.hadoop.hive.ql.exec.mr.MapRedTask
```

错误表现 2：

```
1.  <property>
2.    <name>hive.aux.jars.path</name>
3.    <value>
4.  file:///usr/hive/lib/hive-hbase-handler-0.13.0-SNAPSHOT.jar,
5.  file:///usr/hive/lib/protobuf-java-2.5.0.jar,
6.  file:///usr/hive/lib/hbase-client-0.96.0-hadoop2.jar,
7.  file:///usr/hive/lib/hbase-common-0.96.0-hadoop2.jar,
8.  file:///usr/hive/lib/zookeeper-3.4.5.jar,
9.  file:///usr/hive/lib/guava-11.0.2.jar</value>
10. </property>
11. <property>
```

上面看那上去很整洁，但是如果直接复制到配置文件中，就会产生下面错误。

```
1.  Caused by: java.net.URISyntaxException: Illegal character in scheme name
    at index 0:
2.  file:///usr/hive/lib/protobuf-java-2.5.0.jar
3.          at java.net.URI$Parser.fail(URI.java:2829)
4.          at java.net.URI$Parser.checkChars(URI.java:3002)
5.          at java.net.URI$Parser.checkChar(URI.java:3012)
6.          at java.net.URI$Parser.parse(URI.java:3028)
7.          at java.net.URI.<init>(URI.java:753)
8.          at org.apache.hadoop.fs.Path.initialize(Path.java:203)
9.          ... 37 more
```

```
10. Job Submission failed with exception
    'java.lang.IllegalArgumentException(java.net.URISyntaxException: Illegal
    character in scheme name at index 0:
11. file:///usr/hive/lib/protobuf-java-2.5.0.jar)'
12. FAILED: Execution Error, return code 1 from
    org.apache.hadoop.hive.ql.exec.mr.MapRedTask
```

验证 hive 与 hbase 的整合：

一、启动 hbase 与 hive

启动 hbase

```
1.  hbase shell
```

复制代码

```
aboutyun@master:~$ start-hbase.sh
slave1: starting zookeeper, logging to /usr/hbase/bin/../logs/hbase-aboutyu
keeper-slave1.out
slave2: starting zookeeper, logging to /usr/hbase/bin/../logs/hbase-aboutyu
keeper-slave2.out
master: starting zookeeper, logging to /usr/hbase/bin/../logs/hbase-aboutyu
keeper-master.out
starting master, logging to /usr/hbase/bin/../logs/hbase-aboutyun-master-ma
out
slave2: starting regionserver, logging to /usr/hbase/bin/../logs/hbase-abou
regionserver-slave2.out
slave1: starting regionserver, logging to /usr/hbase/bin/../logs/hbase-abou
regionserver-slave1.out
```

启动 hive

(1)启动元数据库

```
aboutyun@master:~$ hive --service metastore
Starting Hive Metastore Server
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.input.dir.recursiv
deprecated. Instead, use mapreduce.input.fileinputformat.input.dir.recursiv
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.max.split.size is
cated. Instead, use mapreduce.input.fileinputformat.split.maxsize
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.min.split.size is
cated. Instead, use mapreduce.input.fileinputformat.split.minsize
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.min.split.size.per
 is deprecated. Instead, use mapreduce.input.fileinputformat.split.minsize.
ack
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.min.split.size.per
 is deprecated. Instead, use mapreduce.input.fileinputformat.split.minsize.
ode
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.reduce.tasks is de
ted. Instead, use mapreduce.job.reduces
14/05/28 13:37:58 INFO Configuration.deprecation: mapred.reduce.tasks.specu
e.execution is deprecated. Instead, use mapreduce.reduce.speculative
```

```
1.  CREATE TABLE hbase_table_1(key int, value string) STORED BY
    'org.apache.hadoop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES
    ("hbase.columns.mapping" = ":key,cf1:val") TBLPROPERTIES
    ("hbase.table.name" = "xyz");
```

上面的含义是在 hive 中建表 hbase_table_1，通过 org.apache.hadoop.hive.hbase.HBaseStorageHandler 这个类映射，在 hbase 建立与之对应的 xyz 表。

（1）执行这个语句之前：

首先查看 hbase 与 hive：

hbase 为空：

```
hbase(main):023:0> list
TABLE
0 row(s) in 0.0940 seconds

=> []
hbase(main):024:0>
```

hive 为空

```
hive> show tables;
OK
Time taken: 0.039 seconds
```

（2）执行

```
1.  CREATE TABLE hbase_table_1(key int, value string) STORED BY
    'org.apache.hadoop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES
    ("hbase.columns.mapping" = ":key,cf1:val") TBLPROPERTIES
    ("hbase.table.name" = "xyz");
```

```
hive> CREATE TABLE hbase_table_1(key int, value string) STORED BY 'org.apache.ha
doop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES ("hbase.columns.mappin
g" = ":key,cf1:val") TBLPROPERTIES ("hbase.table.name" = "xyz");
OK
Time taken: 3.776 seconds
```

（3）对比发生变化

hbase 显示新建表 xyz

```
hbase(main):024:0> list
TABLE
xyz
1 row(s) in 0.0690 seconds

=> ["xyz"]
```

hive 显示新建表 hbase_table_1

```
hive> show tables;
OK
hbase_table_1
Time taken: 0.022 seconds, Fetched: 1 row(s)
hive>
```

三、验证整合，在 **hbase** 插入表

（1）通过 hive 添加数据

在 hbase 中插入一条记录：

```
1.  put 'xyz','10001','cf1:val','www.aboutyun.com'
```

复制代码

```
hbase(main):027:0>  put 'xyz','10001','cf1:val','www.aboutyun.com'
```

分别查看 hbase 与 hive 表发生的变化：

（1）hbase 变化

```
hbase(main):029:0> scan 'xyz'
ROW                           COLUMN+CELL
 10001                        column=cf1:val, timestamp=1401258963479, value=
1 row(s) in 0.0530 seconds
```

（2）hive 变化

```
hive> select * from hbase_table_1;
OK
10001   www.aboutyun.com
Time taken: 3.498 seconds, Fetched: 1 row(s)
hive>
```

（2）通过 hbase 添加数据

对于网上流行的通过 pokes 表，插入这里没有执行成功，通过网上查询，可能是 hive0.12 的一个 bug.详细可以查看：

```
1.  INSERT OVERWRITE TABLE hbase_table_1 SELECT * FROM pokes;
2.  Total MapReduce jobs = 1
3.  Launching Job 1 out of 1
4.  Number of reduce tasks is set to 0 since there's no reduce operator
5.  java.lang.IllegalArgumentException: Property value must not be null
6.  at
    com.google.common.base.Preconditions.checkArgument(Preconditions.java:8
    8)
7.  at org.apache.hadoop.conf.Configuration.set(Configuration.java:810)
```

```
8.  at org.apache.hadoop.conf.Configuration.set(Configuration.java:792)
9.  at
    org.apache.hadoop.hive.ql.exec.Utilities.copyTableJobPropertiesToConf(U
    tilities.java:1996)
10. at
    org.apache.hadoop.hive.ql.exec.FileSinkOperator.checkOutputSpecs(FileSi
    nkOperator.java:864)
11. at
    org.apache.hadoop.hive.ql.io.HiveOutputFormatImpl.checkOutputSpecs(Hive
    OutputFormatImpl.java:67)
12. at
    org.apache.hadoop.mapreduce.JobSubmitter.checkSpecs(JobSubmitter.java:4
    58)
13. at
    org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter
    .java:342)
14. at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1268)
15. at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1265)
16. at java.security.AccessController.doPrivileged(Native Method)
17. at javax.security.auth.Subject.doAs(Subject.java:415)
18. at
    org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformati
    on.java:1491)
19. at org.apache.hadoop.mapreduce.Job.submit(Job.java:1265)
20. at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:562)
21. at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:557)
22. at java.security.AccessController.doPrivileged(Native Method)
23. at javax.security.auth.Subject.doAs(Subject.java:415)
24. at
    org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformati
    on.java:1491)
```

```
25. at
    org.apache.hadoop.mapred.JobClient.submitJobInternal(JobClient.java:557
    )
26. at org.apache.hadoop.mapred.JobClient.submitJob(JobClient.java:548)
27. at
    org.apache.hadoop.hive.ql.exec.mr.ExecDriver.execute(ExecDriver.java:42
    4)
28. at
    org.apache.hadoop.hive.ql.exec.mr.MapRedTask.execute(MapRedTask.java:13
    6)
29. at org.apache.hadoop.hive.ql.exec.Task.executeTask(Task.java:152)
30. at
    org.apache.hadoop.hive.ql.exec.TaskRunner.runSequential(TaskRunner.java
    :65)
31. at org.apache.hadoop.hive.ql.Driver.launchTask(Driver.java:1481)
32. at org.apache.hadoop.hive.ql.Driver.execute(Driver.java:1258)
33. at org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1092)
34. at org.apache.hadoop.hive.ql.Driver.run(Driver.java:932)
35. at org.apache.hadoop.hive.ql.Driver.run(Driver.java:922)
36. at
    org.apache.hadoop.hive.cli.CliDriver.processLocalCmd(CliDriver.java:268
    )
37. at org.apache.hadoop.hive.cli.CliDriver.processCmd(CliDriver.java:220)
38. at org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:422)
39. at
    org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:790)
40. at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:684)
41. at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:623)
42. at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
43. at
    sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.ja
    va:57)
```

```
44. at
    sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccesso
    rImpl.java:43)
45. at java.lang.reflect.Method.invoke(Method.java:606)
46. at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
47. Job Submission failed with exception
    'java.lang.IllegalArgumentException(Property value must not be null)'
48. FAILED: Execution Error, return code 1 from
    org.apache.hadoop.hive.ql.exec.mr.MapRedTask
```

网上找了很多资料，这个可能是一个 bug，在 hive0.13.0 已经修复。
详细见：
https://issues.apache.org/jira/browse/HIVE-5515

www.aboutyun.com

欢迎加入 about 云官方群 **39327136**、**371358502**
云计算爱好者群

关注微信：

云资源、云技术、疑问解答邮件订阅地址：
http://list.qq.com/cgi-bin/qf_invite?id=a52796b9458ddfda62e055bf899ec187af5860bbdcbf5ec9

淘宝云技术入门、hadoop、openstack 及其它视频：
http://aboutyun.taobao.com/?v=1