一、准备工作

找一台Linux主机，由于spark源码编译会下载很多的第三方类库包，因此需要主机能够联网。

1、安装Java，配置环境变量，版本为JDK1.7或者以上

下载地址：http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html

```
export JAVA_HOME=/usr/java/default
export JRE_HOME=/usr/java/default/jre
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar:$JRE_HOME/lib:$CLASSPATH
export PATH=$JAVA_HOME/bin:$PATH
```

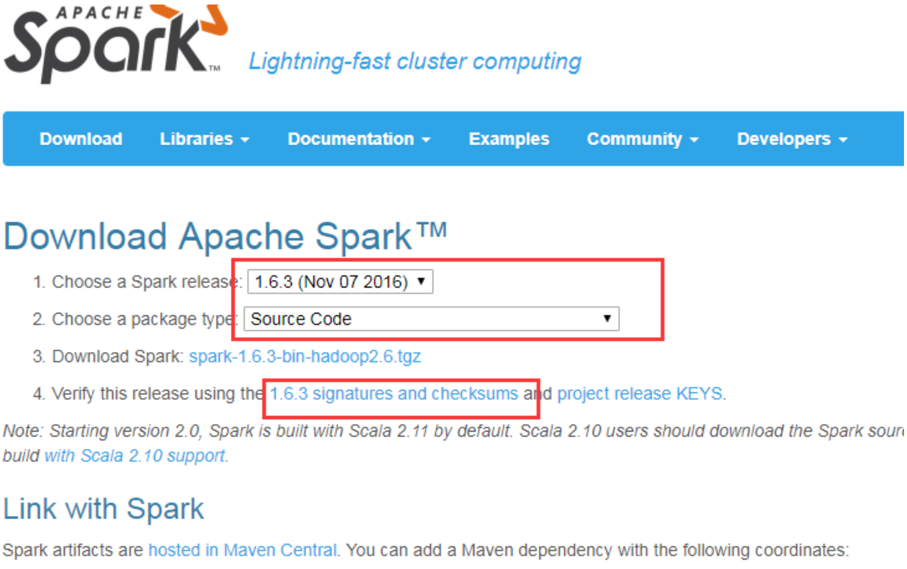
2、安装Maven，版本为3.3.9或者以上

下载地址：https://mirrors.tuna.tsinghua.edu.cn/apache//maven/maven-3/3.3.9/binaries/

```
export MAVEN_HOME=/usr/local/apache-maven-3.3.9
export PATH=$MAVEN_HOME/bin:$PATH
```

二、编译Spark的源码包

1、下载spark 1.6.3 的源码包



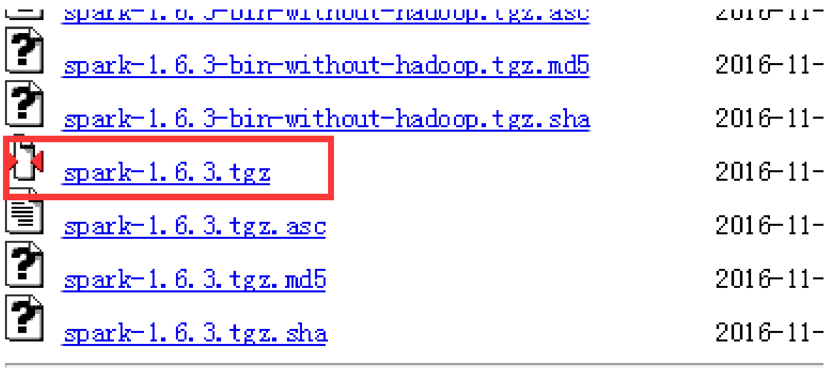
Download Apache Spark™

1. Choose a Spark release: 1.6.3 (Nov 07 2016)
2. Choose a package type: Source Code
3. Download Spark: spark-1.6.3-bin-hadoop2.6.tgz
4. Verify this release using the 1.6.3 signatures and checksums and project release KEYS.

Note: Starting version 2.0, Spark is built with Scala 2.11 by default. Scala 2.10 users should download the Spark sour build with Scala 2.10 support.

Link with Spark

Spark artifacts are hosted in Maven Central. You can add a Maven dependency with the following coordinates:

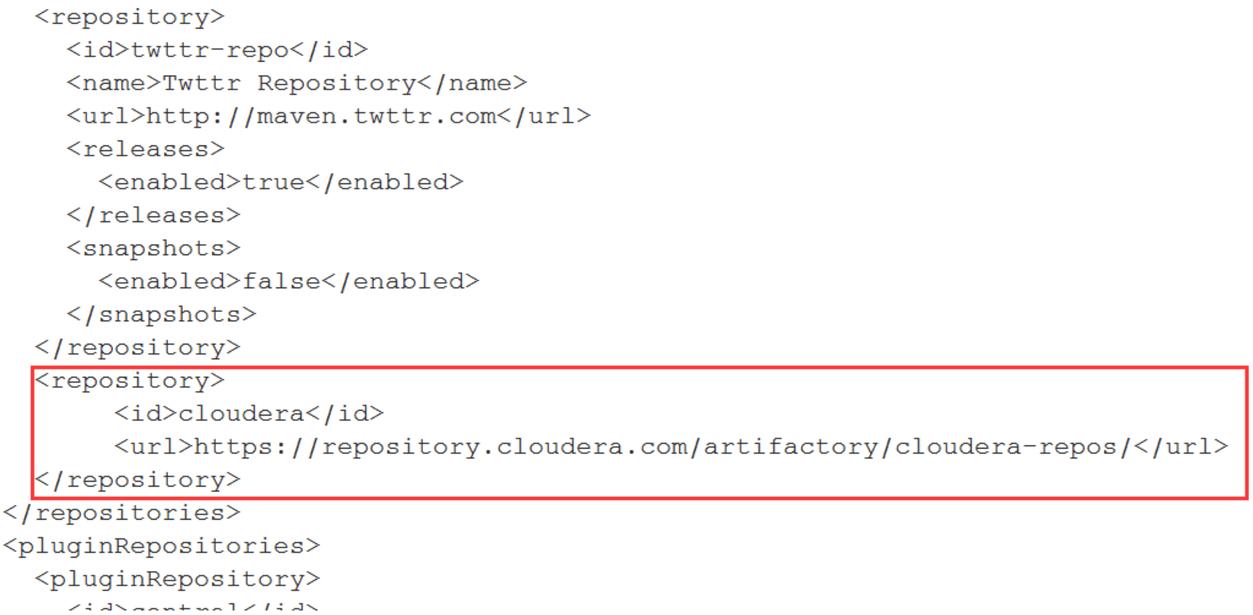


```
spark-1.6.3-bin-without-hadoop.tgz.asc    2016-11-
spark-1.6.3-bin-without-hadoop.tgz.md5    2016-11-
spark-1.6.3-bin-without-hadoop.tgz.sha    2016-11-
spark-1.6.3.tgz                           2016-11-
spark-1.6.3.tgz.asc                       2016-11-
spark-1.6.3.tgz.md5                       2016-11-
spark-1.6.3.tgz.sha                       2016-11-
```

2、增加cdh的repository

解压spark的源码包，编辑pom.xml文件， 在repositories节点 加入如下配置：

```xml
    <repository>
        <id>cloudera</id>
        <url>https://repository.cloudera.com/artifactory/cloudera-repos/</url>
    </repository>
```



```
    <repository>
      <id>twttr-repo</id>
      <name>Twttr Repository</name>
      <url>http://maven.twttr.com</url>
      <releases>
        <enabled>true</enabled>
      </releases>
      <snapshots>
        <enabled>false</enabled>
      </snapshots>
    </repository>
    <repository>
         <id>cloudera</id>
         <url>https://repository.cloudera.com/artifactory/cloudera-repos/</url>
    </repository>
  </repositories>
  <pluginRepositories>
    <pluginRepository>
      <id>central</id>
```

3、开始编译

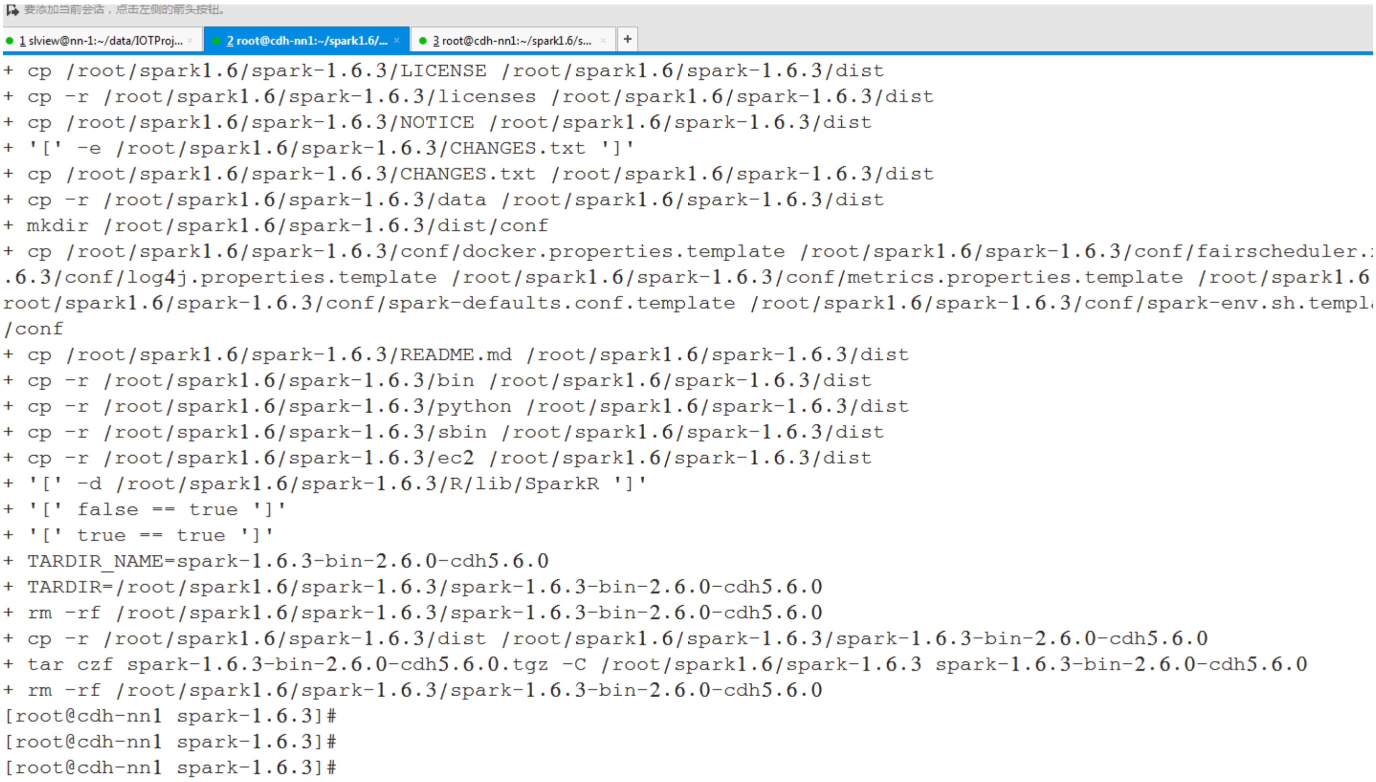
```
./make-distribution.sh --name 2.6.0-cdh5.6.0 --tgz  -Pyarn -Phadoop-2.6 -Phive -Phive-thriftserver -Dhadoop.version=2.6.0-cdh5.6.0
```

如果需要对scala2.11支持：

```
./make-distribution.sh --name 2.6.0-cdh5.6.0 --tgz  -Pyarn -Phadoop-2.6 -Phive -Phive-thriftserver -Dscala-2.11 -Dhadoop.version=2.6.0-cdh5.6.0
```

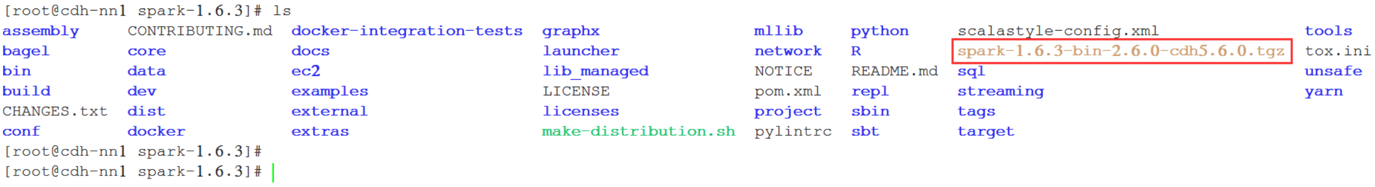
在编译过程中， 可能会出现各种莫名其妙的原因导致中断， 只需要重新执行上面的编译命令即可， 第一次编译可能需要几个小时，第一次编译成功后， 后面再编译就很快了。

编译成功后， 可以看到如下：


要添加当前会话，点击左侧的箭头按钮。
1 slview@nn-1:~/data/IOTProj...
2 root@cdh-nn1:~/spark1.6/...
3 root@cdh-nn1:~/spark1.6/s...
+

```
+ cp /root/spark1.6/spark-1.6.3/LICENSE /root/spark1.6/spark-1.6.3/dist
+ cp -r /root/spark1.6/spark-1.6.3/licenses /root/spark1.6/spark-1.6.3/dist
+ cp /root/spark1.6/spark-1.6.3/NOTICE /root/spark1.6/spark-1.6.3/dist
+ '[' -e /root/spark1.6/spark-1.6.3/CHANGES.txt ']'
+ cp /root/spark1.6/spark-1.6.3/CHANGES.txt /root/spark1.6/spark-1.6.3/dist
+ cp -r /root/spark1.6/spark-1.6.3/data /root/spark1.6/spark-1.6.3/dist
+ mkdir /root/spark1.6/spark-1.6.3/dist/conf
+ cp /root/spark1.6/spark-1.6.3/conf/docker.properties.template /root/spark1.6/spark-1.6.3/conf/fairscheduler.
.6.3/conf/log4j.properties.template /root/spark1.6/spark-1.6.3/conf/metrics.properties.template /root/spark1.6
root/spark1.6/spark-1.6.3/conf/spark-defaults.conf.template /root/spark1.6/spark-1.6.3/conf/spark-env.sh.templa
/conf
+ cp /root/spark1.6/spark-1.6.3/README.md /root/spark1.6/spark-1.6.3/dist
+ cp -r /root/spark1.6/spark-1.6.3/bin /root/spark1.6/spark-1.6.3/dist
+ cp -r /root/spark1.6/spark-1.6.3/python /root/spark1.6/spark-1.6.3/dist
+ cp -r /root/spark1.6/spark-1.6.3/sbin /root/spark1.6/spark-1.6.3/dist
+ cp -r /root/spark1.6/spark-1.6.3/ec2 /root/spark1.6/spark-1.6.3/dist
+ '[' -d /root/spark1.6/spark-1.6.3/R/lib/SparkR ']'
+ '[' false == true ']'
+ '[' true == true ']'
+ TARDIR_NAME=spark-1.6.3-bin-2.6.0-cdh5.6.0
+ TARDIR=/root/spark1.6/spark-1.6.3/spark-1.6.3-bin-2.6.0-cdh5.6.0
+ rm -rf /root/spark1.6/spark-1.6.3/spark-1.6.3-bin-2.6.0-cdh5.6.0
+ cp -r /root/spark1.6/spark-1.6.3/dist /root/spark1.6/spark-1.6.3/spark-1.6.3-bin-2.6.0-cdh5.6.0
+ tar czf spark-1.6.3-bin-2.6.0-cdh5.6.0.tgz -C /root/spark1.6/spark-1.6.3 spark-1.6.3-bin-2.6.0-cdh5.6.0
+ rm -rf /root/spark1.6/spark-1.6.3/spark-1.6.3-bin-2.6.0-cdh5.6.0
[root@cdh-nn1 spark-1.6.3]#
[root@cdh-nn1 spark-1.6.3]#
[root@cdh-nn1 spark-1.6.3]#
```

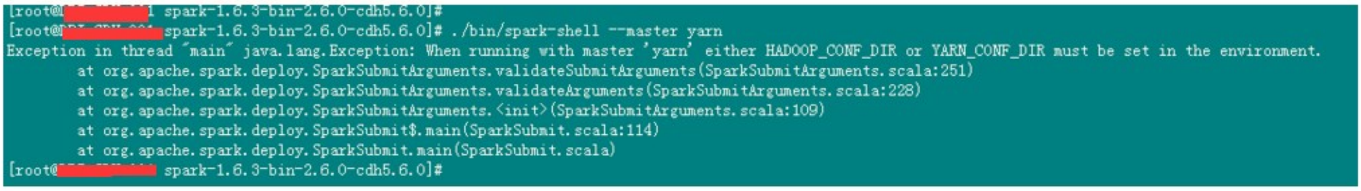
编译成功后，可以看到生成了tar包：



```
[root@cdh-nn1 spark-1.6.3]# ls
assembly      CONTRIBUTING.md  docker-integration-tests  graphx                mllib     python     scalastyle-config.xml                tools
bagel         core             docs                      launcher              network   R          spark-1.6.3-bin-2.6.0-cdh5.6.0.tgz   tox.ini
bin           data             ec2                       lib_managed           NOTICE    README.md  sql                                  unsafe
build         dev              examples                  LICENSE               pom.xml   repl       streaming                            yarn
CHANGES.txt   dist             external                  licenses              project   sbin       tags
conf          docker           extras                    make-distribution.sh  pylintrc  sbt        target
[root@cdh-nn1 spark-1.6.3]#
[root@cdh-nn1 spark-1.6.3]#
```

三、测试

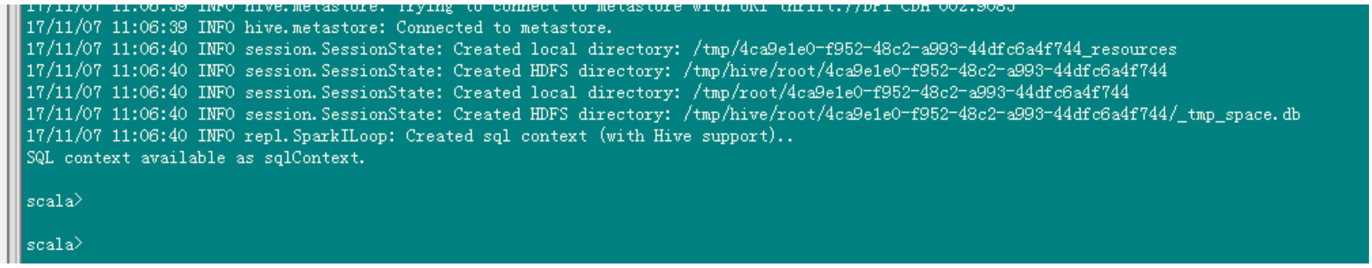
1、提交到yarn上面



```
[root@[redacted] spark-1.6.3-bin-2.6.0-cdh5.6.0]#
[root@[redacted]spark-1.6.3-bin-2.6.0-cdh5.6.0]# ./bin/spark-shell --master yarn
Exception in thread "main" java.lang.Exception: When running with master 'yarn' either HADOOP_CONF_DIR or YARN_CONF_DIR must be set in the environment.
        at org.apache.spark.deploy.SparkSubmitArguments.validateSubmitArguments(SparkSubmitArguments.scala:251)
        at org.apache.spark.deploy.SparkSubmitArguments.validateArguments(SparkSubmitArguments.scala:228)
        at org.apache.spark.deploy.SparkSubmitArguments.<init>(SparkSubmitArguments.scala:109)
        at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:114)
        at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
[root@[redacted] spark-1.6.3-bin-2.6.0-cdh5.6.0]#
```

需要配置HADOOP_CONF_DIR或者YARN_CONF_DIR环境变量：

```
# export HADOOP_CONF_DIR=/etc/hadoop/conf
```

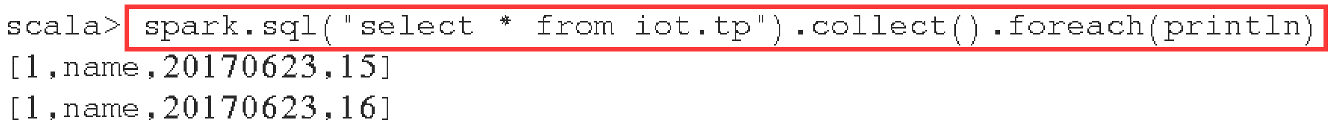


```
17/11/07 11:06:39 INFO hive.metastore: Trying to connect to metastore with URI thrift://DPI-CDH-002:9083
17/11/07 11:06:39 INFO hive.metastore: Connected to metastore.
17/11/07 11:06:40 INFO session.SessionState: Created local directory: /tmp/4ca9e1e0-f952-48c2-a993-44dfc6a4f744_resources
17/11/07 11:06:40 INFO session.SessionState: Created HDFS directory: /tmp/hive/root/4ca9e1e0-f952-48c2-a993-44dfc6a4f744
17/11/07 11:06:40 INFO session.SessionState: Created local directory: /tmp/root/4ca9e1e0-f952-48c2-a993-44dfc6a4f744
17/11/07 11:06:40 INFO session.SessionState: Created HDFS directory: /tmp/hive/root/4ca9e1e0-f952-48c2-a993-44dfc6a4f744/_tmp_space.db
17/11/07 11:06:40 INFO repl.SparkILoop: Created sql context (with Hive support)..
SQL context available as sqlContext.

scala>

scala>
```

```
val file=sc.textFile("/tmp/appveyor.yml")
val wc = file.flatMap(line => line.split(",")).map(word=>(word,1)).reduceByKey(_ + _)
```

2、访问hive的表

需要将hive的hive-site.xml复制到spark的conf目录下面。
scala> spark.sql("select * from iot.tp").collect().foreach(println)

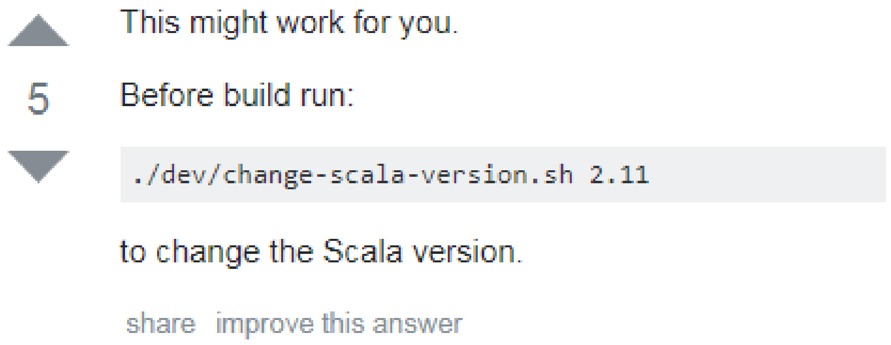


```
scala> spark.sql("select * from iot.tp").collect().foreach(println)
[1,name,20170623,15]
[1,name,20170623,16]
```

编译scala2.11 报错：
报错：
Failed to execute goal net.alchim31.maven:scala-maven-plugin:3.2.2:compile (scala-compile-first) on

执行下面的语句即可：

```
./dev/change-scala-version.sh 2.11
```



This might work for you.

Before build run:

```
./dev/change-scala-version.sh 2.11
```

to change the Scala version.