Big Data
- Hadoop Installation

Dr. Qing “Matt” Zhang
ITU

Create hadoop user and group

- Run the following commands to add a new user “hduser”, with the group “hadoop”
  - *sudo groupadd hadoop*
  - *sudo useradd -g hadoop hduser*
  - *sudo passwd hduser*
    - Set the password to ‘hduser’ in the tutorial
  - *su - hduser*

Passphraseless ssh

- check that you can ssh to the localhost without a passphrase
  - *ssh localhost*
- If cannot ssh to localhost without a passphrase, execute the following commands:
  - *ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa*
    - After -P are two single quotes ’, not double quote “, which means empty password
  - *cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys*

Java Install

- Hadoop requires at least a late version of Java 6
- Check the existing java version and path

```
[hduser@localhost ~]$ which java
/bin/java
[hduser@localhost ~]$ java -version
java version "1.7.0_65"
OpenJDK Runtime Environment (rhel-2.5.1.2.el7_0-x86_64 u65-b17)
OpenJDK 64-Bit Server VM (build 24.65-b04, mixed mode)
```

Java Install (cont'd)

- Suppose you want to download the latest Java
- Download the latest JDK
  - Search for "jdk 1.8" online, the 1st Oracle link return by Google
  - Accept license agreement and download jdk-8u45-linux-x64.tar.gz
- Install it
  - Open another terminal as adminuser, move the tar file you just downloaded to the hduser home directory:
    - *sudo mv ~/Downloads/jdk-8u45-linux-x64.tar.gz /home/hduser*
  - Go back to the hduser window and untar the package
    - *tar -xvf jdk-8u45-linux-x64.tar.gz*

Java Install (cont'd)

- Setup environment in ~/.bash_profile
  - *vi ~/.bash_profile*
  - add the following two lines to this file:
    - *export JAVA_HOME=~/jdk1.8.0_45*
    - *PATH=$JAVA_HOME/bin:$PATH*
- Check setup:

```
[hduser@localhost ~]$ source  ~/.bash_profile
[hduser@localhost ~]$ which java
~/jdk1.8.0_45/bin/java
[hduser@localhost ~]$ java -version
java version "1.8.0_45"
Java(TM) SE Runtime Environment (build 1.8.0_45-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.45-b02, mixed mode)
```

Hadoop Install

- Go to Apache official site and download the latest hadoop tar file
  - http://hadoop.apache.org/releases.html
  - Download hadoop-2.7.0.tar.gz
- Extract it
  - Open a terminal as adminuser, and move the tar file to home directory of hduser
    - *sudo mv ~/Downloads/hadoop-2.7.0.tar.gz /home/hduser*
  - open another terminal su as hduser
    - *tar -xvf hadoop-2.7.0.tar.gz*

Hadoop Install(cont'd)

- Edit ~/.bash_profile and setup environment, add the following two lines:
  - *export HADOOP_PREFIX=~/hadoop-2.7.0*
    - HADOOP_HOME is deprecated and no longer used in Hadoop 2
  - *PATH=$JAVA_HOME/bin:$HADOOP_PREFIX/bin:$PATH*
- Souce the file and make the environment variable effective in your terminal
  - *source ~/.bash_profile*

Hadoop Install(cont'd)

- Setup $JAVA_HOME in $HADOOP_PREFIX/etc/hadoop/hadoop-env.sh
  - Replace and comment out the the existing line as:
  - *#export JAVA_HOME=${JAVA_HOME}*
  - *export JAVA_HOME=~/jdk1.8.0_45*

Test standalone operation

- *cd ~*
- *mkdir input*
- *cp $HADOOP_PREFIX/etc/hadoop/*.xml input*
- *hadoop jar $HADOOP_PREFIX/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0.jar grep input output 'dfs[a-z.]+'*
- *cat output/**

```
[hduser@localhost ~]$ cat output/*
1       dfsadmin
```

Setup Pseudo-distributed operation

- Configure core-site.xml
  - vi $HADOOP_PREFIX/etc/hadoop/core-site.xml

```
<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://localhost:9000</value>
    </property>
</configuration>
```

Setup Pseudo-distributed operation

- Configure hdfs-site.xml
  - vi $HADOOP_PREFIX/etc/hadoop/hdfs-site.xml

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
</configuration>
```

Setup env

- Edit file ~/.bash_profile and setup the environment, add the following two lines:
  - *export HADOOP_COMMON_LIB_NATIVE_DIR= $HADOOP_PREFIX/lib/native*
  - *export HADOOP_OPTS="-Djava.library.path= $HADOOP_PREFIX/lib"*

.bash_profile

- My final .bash_profile looks like:

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs
export JAVA_HOME=~/jdk1.8.0_45
export HADOOP_PREFIX=~/hadoop-2.7.0
PATH=$PATH:$HOME/.local/bin:$HOME/bin
PATH=$JAVA_HOME/bin:$HADOOP_PREFIX/bin:$PATH
export PATH

export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_PREFIX/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_PREFIX/lib"
```

Bring up the system

- Format the file system
  - *hdfs namenode -format*
- Start namenode and datanode
  - *cd $HADOOP_PREFIX*
  - *sbin/start-dfs.sh*
- Check status
  - jps
  - http://localhost:50070

jps output

- You can see the NameNode, DataNode, and SecondaryNameNode are running

```
[hduser@localhost hadoop-2.7.0]$ jps
3840 DataNode
21780 Jps
4104 SecondaryNameNode
3690 NameNode
```

Test run

- Go to hadoop home directory
  - *cd $HADOOP_PREFIX*
- Make an input directory in HDFS:
  - *hadoop fs -mkdir /input*
- copy some random file into input
  - *hadoop fs -put etc/hadoop/*xml /input*
- Run the test jar file
  - *hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.0.jar grep /input /output 'dfs[a-z.]+'*

Test run

- Check the output in HDFS /output directory:

```
[hduser@localhost hadoop-2.7.0]$ hadoop fs -ls /output
Found 2 items
-rw-r--r--   1 hduser supergroup          0 2015-06-04 15:23 /output/_SUCCESS
-rw-r--r--   1 hduser supergroup         29 2015-06-04 15:23 /output/part-r-00000
[hduser@localhost hadoop-2.7.0]$ hadoop fs -cat /output/part-r-00000
1       dfsadmin
1       dfs.replication
```

Test run

- You may see warnings like the following, and you can ignore it:
  - 15/06/19 01:43:21 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
- It's because by default hadoop source code is compiled on a 32-bit platform. If you want to remove it, download the hadoop source and recompile on your own 64-bit platform

Yarn setup

- *cp etc/hadoop/mapred-site.xml.template etc/hadoop/mapred-site.xml*
- *vi etc/hadoop/mapred-site.xml*

```
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
</configuration>
```

Yarn setup

- *vi etc/hadoop/yarn-site.xml*

```
<configuration>
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
</configuration>
```

- Start ResourceManager daemon and NodeManager daemon
  - *sbin/start-yarn.sh*

Verify Yarn Status

```
[hduser@localhost hadoop-2.7.0]$ jps
17952 NameNode
21618 Jps
21555 NodeManager
18116 DataNode
18372 SecondaryNameNode
21419 ResourceManager
```

- You can see that ResourceManager and NodeManager are now running.
- http://localhost:8088/

Quit

- Take a snapshot before quitting the VM