HADOOP INSTALLATION STEPS

1.INSTALL JAVA 1.8.0-171

2.INSTALL UBUNTU 17.10 IN DUAL MODE.

* 1. WHILE INSTALLING REMEMBER THE USERNAME AND PASSWORD. THIS WILL BE UBUNTU USER.

2.1 GET UPDATED VERSION OF UBUNTU, IF ALREADY UBUNTU IS INSTALLED IN THE SYSTEM.

kamala$ sudo apt-get install update [kamala is my ubuntu user]

3.CHECK /VERIFY WHETHER JAVA IS WORKING OR NOT.

kamala$ javac

3.1 CHECK THE VERSION OF JAVA. THIS HAS TO BE GIVEN WHILE CONFIGURING HADOOP

ENVIRONMENT.

kamala$ java -version

4. CREATE ONE GROUP FOR HADOOP

kamala$ sudo addgroup hadoop

5. CREATE USER FOR THE ABOVE CREATED HADOOP GROUP

kamala$ sudo adduser --ingroup <groupname> <username>

kamala$ sudo adduser --ingroup kamala kkhadoopuser

5.1 NOW HERE FOR THIS GIVE THE HADOOP PASSWORD FOR AUTHENTICATION

6. GIVE SUPER USER PERMISSIONS TO HADOOP USER

kamala$ sudo adduser kkhadoopuser sudo

7. INSTALL SECURE SHELL FOR CLIENT AND SERVER OR SERVER ONLY FOR SECURITY

kamala$ sudo apt-get install openssh.server

8. ENTER/LOGIN IN TO TERMINAL WITH HADOOP USER CREATED IN STEP 5.

kamala$ su kkhadoopuser

8.1 GIVE THE PASSWORD WHICH IS GIVEN IN STEP 5.1

9. CHECK THE PRESENT WORKING DIRECTORY

kamala$ pwd

10. GENERATE PUBLIC KEY (SSH KEY) BY USING RSA ALGORITHM(Public **key** authentication is an alternative means of identifying yourself to a login server, instead of typing a password. ...)

$ ssh-keygen -t rsa -P “ ”

11. ENTER FILE IN WHICH TO SAVE THE KEY (/home/kkhadoopuser/.ssh/id\_rsa): Press Enter Key

Note : Press y or yes when prompt messages required

kkhadoopuser$cat $HOME/.ssh/id\_rsa.pub >> $HOME/.ssh/authorized\_keys

12. RUN SSH LOCALHOST

kkhadoopuser$ ssh localhost

Are you sure you want to continue connecting(yes/no):yes

13. NOW, EXIT FROM LOCALHOST

kkhadoopuser$ exit

Now, Download and Extract Hadoop software:

wget is non-interactive tool to download files from web.

Otherwise goto step 3, if it is already downloaded and in downloads.

1. kkhadoopuser$ wget <http://mirrors.sonic.net/apache/hadoop/common/>[hadoop-2.9.1/hadoop-](http://mirrors.sonic.net/apache/hadoop/common/hadoop-2.7.1/hadoop-)2.9.1.tar.gz
2. kkhadoopuser$ cd Downloads

Now Extract Hadoop. tar command is equals to Unzip in Windows

x- extract , v – verbose , z – zip/unzip , and f – file

1. [kamala@Satellite](mailto:3.kamala@Satellite) $ /home/Downloads $ tar xvzf hadoop- 2.9.1.tar.gz

While extracting hadoop folder which is in hadoop-2.9.1 will be extracted and we need to give the path where to be extracted.

4.Move hadoop from Downloads folder into /usr/local/ folder using mv command :

kkhadoopuser$ sudo mv hadoop /usr/local/

5.Change Directory to back/previous directory

kkhadoopuser$ cd

6.Change Owner permissions of hadoop user

kkhadoopuser$ sudo chown -R kkhadoopuser /usr/local

Configuration of Environments for Hadoop

(Setting Various Paths):

Note: nano,gedit are default editors of Ubuntu OS.

1. kkhadoopuser$ sudo nano ~/.bashrc

ADD the following lines of code at the end of the .bashrc file

(AFTER fi statement at the end of the file )

copy:

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64 /Your Java edition

export HADOOP\_HOME=/usr/local/hadoop

export PATH=$PATH:$HADOOP\_HOME/bin

export PATH=$PATH:$JAVA\_HOME/bin

export PATH=$PATH:$HADOOP\_HOME/sbin

export HADOOP\_MAPRED\_HOME=$HADOOP\_HOME

export HADOOP\_COMMON\_HOME=$HADOOP\_HOME

export HADOOP\_HDFS\_HOME=$HADOOP\_HOME

export YARN\_HOME=$HADOOP\_HOME

export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_HOME/lib/native

export PATH=$PATH:$HADOOP\_HOME/sbin:$HADOOP\_HOME/bin

export HADOOP\_INSTALL=$HADOOP\_HOME

export HADOOP\_LOG\_DIR=$HADOOP\_HOME/logs

export HADOOP\_OPTS="$HADOOP\_OPTS -Djava.net.preferIPv4Stack=true -Djava.library.path=$HADOOP\_HOME/lib"

export LD\_LIBRARY\_PATH=/usr/lib/hadoop/lib/native

and paste into .bashrc file which is opened above.

Then to save and close bashrc file :

1.Press CTRL+O 2. Press Enter Key 3.Press CTRL +X

1. Execute script file

kkhadoopuser$ source ~/.bashrc

1. Edit hadoop-env.sh shell file

[hdvsm](mailto:hdvsm@vsmcoesys058)$sudo nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh

Find JAVA\_HOME and change JAVA\_HOME= path line with the following 3 lines

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64

export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_HOME/lib

export HADOOP\_OPTS="$HADOOP\_OPTS - D java.library.path=$HADOOP\_HOME/lib"

1.Press CTRL+O 2. Press Enter Key 3.Press CTRL +X

1. Edit core-site.xml file:

kkhadoopuser$ sudo nano /usr/local/hadoop/etc/hadoop/core-site.xml

Paste the following property code between

<configuration> </configuration> tag:

<property>

<name>fs.default.name</name>

<value>hdfs://localhost:9000</value>

</property>

1.Press CTRL+O 2. Press Enter Key 3.Press CTRL +X

1. Now Create Folder for HDFS for maintaining NameNode and DataNode

kkhadoopuser$ sudo mkdir -p /usr/local/hadoop\_tmp

Note: We can give any name as Hadoop folder(i.e., hadoop\_tmp) it shoud be equal to properties of hdfs.site.xml (see hdfs.site.xml )

6.Create Namenode folder in HDFS

kkhadoopuser$ sudo mkdir -p /usr/local/hadoop\_tmp/hdfs/namenode

7.Create Datanode folder in HDFS

kkhadoopuser$ sudo mkdir -p /usr/local/hadoop\_tmp/hdfs/datanode

8.Edit hdfs-site.xml file:

kkhadoopuser$sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml

Paste the following proprty code between

<configuration> </configuration> tag:

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>file:/usr/local/hadoop\_tmp/hdfs/namenode</value>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value>file:/usr/local/hadoop\_tmp/hdfs/datanode</value>

</property>

1.Press CTRL+O 2. Press Enter Key 3.Press CTRL +X

Note: By default,HDFS maintains 3 replications . I have given 1 replication . We can change the value of the replications

9.Edit yarn-site.xml file:

kkhadoopuser$sudo nano /usr/local/hadoop/etc/hadoop/yarn-site.xml

Paste the following property code between

<configuration> </configuration> tag:

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.aux-services.mapreduce\_shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

1.Press CTRL+O 2. Press Enter Key 3.Press CTRL +X

10.Copy mapred-site.xml.template to mapred-site.xml

kkhadoopuser$ cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml

Note:Copy and Paste above 2 lines command as a single command

11.Edit mapred-site.xml file

kkhadoopuser $ sudo nano /usr/local/hadoop/etc/hadoop/mapred-site.xml

Paste the following property code between <configuration>

</configuration> tag

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

1.Press CTRL+O 2. Press Enter Key 3.Press CTRL +X

12.Assign Owner Permissions to HDFS Folder .

kkhadoopuser $sudo chown -R kkhadoopuser /usr/local/hadoop\_tmp

13.Close terminal

14. Open Terminal( Press Ctrl+Alt+T)

15.Login into/with Hadoop username

kamala$ su kkhadoopuser

Password: Type password of hadoop user.

16.Format Name Node of Hadoop Distributed File System (HDFS)

kkhadoopuser$ hdfs namenode -format

Note:Press y or yes when prompt messages required

17.Start Daemons(BackGround processes) Name Node and Data Node,Secondary Name Node, Job Tracker(Node Manager), Task Tracker( Resource Manager)

kkhadoopuser$ start-dfs.sh

Note: some times , system asks as follows for 3 times:

Enter passphrase for key '/home/kkhadoopuser/.ssh/id\_rsa': Press enter Key

hduser@localhost's password: Type password of hadoop user.

18.Start yarn.

kkhadoopuser$ start-yarn.sh

Note: some times , system asks as follows for 2 times:

Enter passphrase for key '/home/kkhadoopuser/.ssh/id\_rsa': Press enter Key

hduser@localhost's password: Type password of hadoop user.

OR you can make all daemons to start

19. $ start-all.sh

20.Monitor JVM Process Status using jps tool

kkhadoopuser$ **jps**

Picked up JAVA\_TOOL\_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar

4615 SecondaryNameNode

5081 NodeManager

4766 ResourceManager

4419 DataNode

5193 Jps

4262 NameNode

Hadoop is installed Successfully

To run any Hadoop commands or/and run MapReduce programs , All Daemons should be activated/started.

21.Close all Daemons at the end the processing of HDFS & MapReduce

kkhadoopuser$ stop-all.sh

----------------------------------------------------------------------------

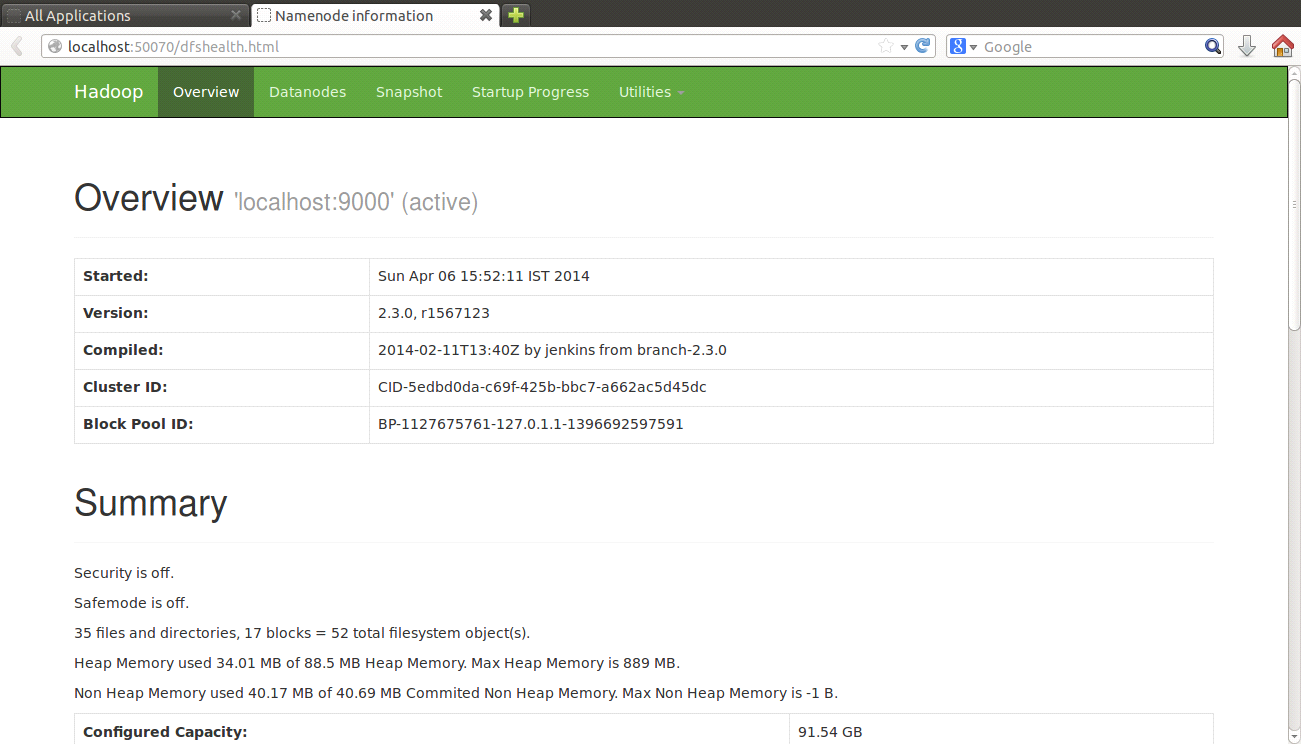
**Opening Cluster User Interfeces(UI) in Browsers**

**To open HDFS and folders in Browser:**

**1. Open any Browser**

**2. Type <http://localhost:50070> in URL address bar**

**3.Click Utilities Menu to Browse HDFS folders and files**



Install and Run Eclipse

1. Open Our Softwares folder in /Home Directory…where eclipse is.

2. Right Click on eclipse-jee-luna-SR2-linux-gtk.-.jar.gz file

3. Click on “Open with Archive manager”

4. Right Click on Eclipse folder

5. Click Extract

6. Click /Home directory in left side Explorer

7. Click Extract button available at Right Side Down Corner.

8. Close Archive Manager Dialog Box. Close Extracted Window

9. Open Terminal (Ctrl + Alt +T)

10. $ sudo mv eclipse /opt/

11. $ sudo chmod 777 /usr/share/applications/

12. $ gedit /usr/share/applications/eclipse.desktop

Copy and Paste the following code:

[Desktop Entry]

Name=EclipseLuna

Name[en]=EclipseLuna

Comment=Integrated Developement Environment

Type=Application

Exec=/opt/eclipse/eclipse

Icon=/opt/eclipse/icon.xpm

Terminal=false

NoDisplay=false

Categories=Development;IDE

Click Save Icon and close gedit window

Now, Eclipse Icon is Created in Ubuntu.

Opening Eclipse

13. To open Eclipse, Click First (Search)Icon/button(Ubuntu Button) of Ubuntu Icons

available in left side .

14. Type Eclipse word in search text box

It shows Eclipse Icon in Applications Group

15. Click Eclipse Icon

Now the system opens Eclipse IDE.

Develop and Run Hadoop MapReduce program using Java in Eclipse

1. Open Terminal and Login into hadoop user.

2. Run all Daemons of Hadoop( start-all.sh and jps commands)

3. Create folder to store our inputfiles in Hadoop HDFS

kkhadoopuser$ hdfs dfs -mkdir /inputs

Note: inputs folder is created in HDFS File System

4. Copy input files into HDFS. The input files may be existing huge files or create new input file for testing purpose. Here Words.txt is a input text file.

4.1: $ sudo nano Words.txt

Type the following words as follows

I am I am I was I can I can able

Press Ctrl+o Press Enter key Press Ctrl +x

Now Copy Local input file into HDFS System

4.2: $ hdfs dfs -put Words.txt /inputs/Words.txt

Now, The file “Words.txt” is copied from Local Directory of Ubuntu into /inputs folder of HDFS.

5. Minimize the Terminal (Do not Close terminal)

-------------------------------------------------------------------------------------

6. Open Eclipse

Note: Create workspace for our project in any folder/directory. Remember workspacename and its directory.Ex: WorkSpaceWordCount. Click Next

7. Once Eclipse is opened, Maximize Eclipse window to display Menu bar.

Otherwise, Menu Bar will not displayed. OR Click New Icon.

8. To open New Project:

8.1. File -> New -> Project -> Java project

8.2. Type projectName Ex: prjWordCount

8.3. Click Next -> Finish

9. Add New Class to the project

9.1. Right Click on Projectname in left side project explorer

9.2. New --> Class ->

9,3.Remove Package name ,if exists. Then class will be stored in default package.and

9.4.Type Class name , which is equals to main () method's class name in your source code

9.5. Click Finish Button

Note: Click OK Button for Perspective message box

Note: if package name is specified when creating class, then

package packagename; should be the first statement of source code of your class. And Use your class as packagename.classname where ever required. But for first examples we can uncheck packages.

10. Copy WordCount.java source code from java file.

11. Paste the code into Eclipse code area and Save project Ctrl+S

12 To Add Hadoop , MapReduce jar files

12.1. Right click on Projectname to get Popup Menu

12.2. Click Build Path --> Add External Archives

12.3. Click FileSystem(Available at leftside) --> usr --> local -->hadoop

-->share --> hadoop -->common -->

12.4 Select all(3) jar files by pressing Shift key and Click OK

12.5. Click FileSystem(Available at leftside) --> usr --> local -->hadoop

-->share --> hadoop -->mapreduce -->

12.6. Select all(9) jar files by pressing Shift key and Click OK

13. To Create jar file for our project

13.1. Right click on Projectname to get Popup Menu

12.2. Click Export -->

12.3. Expand > Java --> click JAR file --> Next

Note: Click OK Button for Warning message box

12.4. Type Jar file name (ex: WordCount.jar…same as class name) in JAR Expert Dialog Box

12.5. Click Next --> Finish

Note:By deafult, The jar file can be created in /workspace folder

---------------------------------------------------------------------------------------

13. Maximize Terminal

14. To run MapReduce Program go to hadoop user. Go to home/system and then to workspace.

kkhadoopuser @system-Satellite: /home/system$ cd workspacename

Note: workspacename indicates workspace name of our project, which is created/specified in first dialog box in Eclipse The workspacename folder contains our jar file.

**Now compile the complete file connected with jars. Below is the format.**

hadoopuser…../workspacename $ hadoop jar jarfile DriverClassName /inputfolder/inputfilename /outputfolder

14.2. kkhadoopuser/admin/WorkSpaceWordCount $ hadoop jar WordCount.jar WordCount /inputs/Words.txt /wordcountoutput

where,

jar file : WordCount.jar

class name: WordCount

inputfolder: /inputs (which is already created)

input filename: Words.txt (which is already created)

outputfolder: /wordcountoutput ( It will be created automatically by the HDFS) to store output files , which are produced by The Hadoop. Separate output folder will be created for each Program

To see the output

15. $ hdfs dfs -cat /wordcountoutput/part-r-00000

It shows final output

OR

15. Open Browser

type [http://localhost:50070](http://localhost:50070/)

click Utilities Link

Click Browse the file system then HDFS shows all folders of HDFS.

Click required folder and open part-r-00000 file , download it.

===========