Single Node Hadoop Cluster Setup

1 Learning Objectives

1. Understand the process of Setting up Single Node Hadoop Cluster

2 Download Links

Hadoop 3.1.3: https://drive.google.com/file/d/1hgQAzqUi2HaSRdX1ZnmVXEBjI04gFYHR/ view?usp=sharing

Eclipse Una: https://drive.google.com/file/d/1i7zkg-L8Rs7JGTju9JIewxnyPLE0\_ZEH/ view?usp=sharing

3 Task-IV: Setting up Hadoop

1. Install openJDK-8

# Remove the existing JDK

$ sudo apt remove default-jdk default-jre openjdk-headless

# install openjdk-8

$ sudo apt install openjdk-8-jdk openjdk-8-jre

# Check the installation

$ java -version

openjdk version "1.8.0\_312"

$ javac -version # java compiler version

javac 1.8.0\_312

2. Adding the JDK path to the System PATH variable and informing the changes to OS

# Open ~/.bashrc and add

$ sudo vim ~/.bashrc

#go to the last line and add the following

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

export PATH=$PATH:$JAVA\_HOME (esc + :wq)

##save and exit

## Inform the OS about the modification

$ source ~/.bashrc

1

# Check the System variables by echoing them

$ echo $PATH

$ echo $JAVA\_HOME

3. Add a dedicated user for the HADOOP called as "hadoop" and make the user to be in sudoers group

$ sudo adduser hadoop # adding a new user hadoop

$ sudo usermod -aG sudo hadoop # making the hadoop user as sudoer

4. Once the user is added, login to the user Hadoop to generate the ssh key for passwordless login (hadoop@machinename)

$ sudo su - hadoop # Switch to the hadoop user

$ ssh-keygen -t rsa # Create the SSH key pair

$ cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys # Add the localhost - server + client

$ chmod 0600 ~/.ssh/authorized\_keys # Change permission of the key

# Check the login to localhost using ssh is valid

$ ssh localhost (install ssh if not found, sudo apt-get install openssh-client openssh-server)

# if login successful - then connection as success - Namenode and Datanode can use this tunnel

$ exit

# PS: Mandatory: Once the connection is made, logout from ssh

5. Download the latest binary from the

https://drive.google.com/file/d/1hgQAzqUi2HaSRdX1ZnmVXEBjI04gFYHR/view?usp=sharing on to the download folder

6. Extract it and move it as /usr/local/hadoop as follows,

# Under Downloads folder

$ tar -xvzf hadoop-3.1.4.tar.gz

$ sudo mv hadoop-3.1.4 /usr/local/hadoop # rename and place the folder under /usr/local

7. Setup the dedicated PATH variables for Hadoop as hadoop-java.sh as follows,

# PS: Make sure the user is hadoop

$ sudo vim /etc/profile.d/hadoop\_java.sh

# Add the following lines

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

export HADOOP\_HOME=/usr/local/hadoop

export HADOOP\_HDFS\_HOME=$HADOOP\_HOME

export HADOOP\_MAPRED\_HOME=$HADOOP\_HOME

export YARN\_HOME=$HADOOP\_HOME

export HADOOP\_COMMON\_HOME=$HADOOP\_HOME

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

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

export HADOOP\_OPTS="$HADOOP\_OPTS -Djava.library.path=$HADOOP\_HOME/lib/native" # Save and exit

# Source the file to update the changes to OS

2

$ source /etc/profile.d/hadoop\_java.sh

8. Con rm your hadoop and hdfs version

$ hadoop version

Hadoop 3.1.4

$ hdfs version

9. Change the ownership of /usr/local/hadoop to hadoop user

$ sudo chown -R hadoop:hadoop /usr/local/hadoop

10. Make two directories hdfs and htemp((Supporting Directories) under $HADOOP\_HOME=/usr/local/hadoop has follows,

$ cd $HADOOP\_HOME

$ mkdir hdfs htemp

11. Give the ownership of these two folders to hadoop user as follows,

$ sudo chown -R hadoop:hadoop /usr/local/hadoop/hdfs

$ sudo chown -R hadoop:hadoop /usr/local/hadoop/htemp

12. The following steps are the configuration to be done in the hadoop installation directories (a) Specify JAVA\_HOME in hadoop-env.sh (/usr/local/hadoop/etc/hadoop)

$ cd $HADOOP\_HOME/etc/hadoop

# Edit the hadoop-env.sh and update JAVA\_HOME variable

$ vim hadoop-env.sh

# Add the following line in java implementation

export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64 #(54th line)

# Save and exit

(b) Modify core-site.xml($HADOOP\_HOME/etc/hadoop) to setup web portal for hadoop

$ vim core-site.sh

# Add the following lines to it

<configuration>

<property>

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

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

<description>The default file system URI</description>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>/usr/local/hadoop/htemp</value>

</property>

</configuration>

# Save and Quit

(c) Modify hdfs-site.xml($HADOOP\_HOME/etc/hadoop) to setup namenode and datanode path and replication factor

3

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

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

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

</property>

<property>

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

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

</property>

</configuration>

(d) Configure the mapreduce framework by editing the mapred-site.xml($HADOOP\_HOME/etc/hadby adding the following lines

<configuration>

<property>

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

<value>yarn</value>

</property>

<property>

<name>mapreduce.application.classpath</name>

<value>$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/\*:$HADOOP\_MAPRED\_HO

ME/share/hadoop/mapreduce/lib/\*</value>

</property>

</configuration>

(e) Configure the YARN resource manager by editing the yarn-site.xml

<configuration>

<property>

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

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.env-whitelist</name>

<value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,

CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>

</property>

</configuration>

13. Format the namenode using the command

$ hdfs namenode -format

14. To start the hadoop demons such as Namenode, Datanode, Resource Manager and Secondary Namenode, issue the following command,

$ cd $HADOOP\_HOME/sbin

# First Method : Start individual demons

4

$ ./start-dfs.sh # Starts the HDFS nodes

$ ./start-yarn.sh # Starts the Resource Manager

# Second Method : Start all the demons

$./start-all.sh # Alternatively starts all the nodes

# To stop the demons

$ cd $HADOOP\_HOME/sbin

$ ./stop-all.sh # Stops all the running nodes

15. Check the availability of all the nodes by typing(When hadoop nodes are running) using fol lowing commands,

$ jps

# Default Nodes to run: Hadoop is up and running

12293 Jps

9877 NameNode

10085 DataNode

10953 NodeManager

10590 ResourceManager

10335 SecondaryNameNode

16. Access the Web portal for hadoop management by typing in the following IP address in the browser http://localhost:9870

17. Check the status of the cluster in the link http://localhost:8088

5