INSTALLATION GUIDE OF ONE NODE HADOOP

# Introduction

Apache Hadoop software library (https://hadoop.apache.org/) is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, where each may be prone to failures. (<http://hadoop.apache.org/>)

## Modules in Hadoop:

### Hadoop Common:

The common utilities that support the other Hadoop modules.

### Hadoop Distributed File System (HDFD):

A distributed file system that provides high-throughput access to application data.

### YARN:

A framework for job scheduling and cluster resource management.

### MapReduce:

A YARN-based system for parallel processing of large data sets.

# Installation

Following we will install Hadoop (2.9.2) on a single node with Ubuntu 20.04 LTS.

## Step 1 – Prerequisites:

On the machine, I’ll first upgrade the package list and upgrade the already installed software. This would help us install the updated version of any software.

sudo apt update  
sudo apt upgrade

For Hadoop to be installed we need following list of software installed.

* Java 8
* Ssh
* Rsync

sudo apt install ssh  
sudo apt install rsync

If java is installed please skip this step or otherwise use the below commands.

sudo apt install openjdk-**8**-jdk-headless/bionic-updates –y

OR

sudo apt install openjdk-**8**-jdk-headless –y

Check to see if JAVA\_HOME is already set

echo $JAVA\_HOME

Its /usr/lib/jvm/java-8-openjdk-amd64/jre on this machine. If you get empty try the following steps.

To find out JAVA\_HOME directory, enter the command:

which java

Result gives /usr/bin/java as the installation directory which is the original directory of Java.

To find out where it is pointing to, enter the following command:

ls -l `which java`

lrwxrwxrwx 1 root root 22 June 5 23:45 /usr/bin/java -> /etc/alternatives/java

This shows that path /usr/bin/java is pointing to /etc/alternatives/java

To find out where /etc/alternatives/java is pointing to use the above command with a bit tweaking.

ls -l /etc/alternatives/java

This gives the following path as result

/etc/alternatives/java is pointing towards /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java

Remove the java and bin in the end and this will give you your Java home path.

/usr/lib/jvm/java-8-openjdk-amd64/jre/

To set JAVA\_HOME, use the following command:

echo 'export JAVA\_HOME=YOUR\_JAVA\_HOME' >> ~/.bashrc

This command will append a new line in your bash path file.

Use following command to apply changes

source ~/.bashrc

Confirm by using the following command, to see if the JAVA\_HOME is set:

echo $JAVA\_HOME

/usr/lib/jvm/java-8-openjdk-amd64/jre

if facing issues try closing and reopening the terminal

## Step 2 – Downloading and Setting

To download Hadoop, visit the link (<https://downloads.apache.org/hadoop/common/hadoop-2.9.2/>) and download the ‘hadoop-2.9.2.tar.gz’ pre build version. You may download the other version of Hadoop on the above link but this installation guide is tested with Hadoop 2.9.2

It will by default downloaded in to you ‘~/Download’ folder.

Extract the installation by using the below command:

tar -xvf hadoop-2.9.2.tar.gz

Move the extracted one to the home folder using following command.

mv Hadoop-2.9.2 /home/<username>/

or

mv Hadoop-2.9.2 ~/

cd in the Hadoop folder that just moved in your home directory

do the following to find the path

pwd

Now setup HADOOP\_HOME variable in you .bashrc file by copying that path in below variable

echo 'export HADOOP\_HOME=/home/geek/hadoop-2.9.2' >> ~/.bashrc

execute the following command to make changes in the .bashrc file.

source ~/.bashrc

## Step 3 – Configuration

Follow the below guide lines to make configuration changes.

Add fs.defaultFS property in HADOOP\_HOME/etc/hadoop/core-site.xml

It should be look like as follow

<?xml version="1.0" encoding="UTF-8"?>  
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

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

This property provides HDFS address to our dfs/fs commands If we don’t specify this we need to specify HDFS address in each of dfs/fs command.

Create Name node and data node in you following home directory

mkdir ~/namenode  
mkdir ~/datanode

We are setting this Hadoop installation for single node. To setup replication factor, our path of name node directory and data node directory, we need to edit HADOOP\_HOME/etc/Hadoop/hdfs-site.xml

The hdfs-site.xml should look like as below

<?xml version="1.0" encoding="UTF-8"?>  
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  
<configuration>  
<property>  
<name>dfs.replication</name>  
<value>1</value>  
</property>  
<property>  
<name>dfs.namenode.name.dir</name>  
<value>/home/geek/namenode</value>  
</property>  
<property>  
<name>dfs.datanode.name.dir</name>  
<value>/home/geek/datanode</value>  
</property>  
</configuration>

Replication factor describes how many time the data is replicating in blocks. Setting it 1 would suggest no replication.

Dfs.namenode.name.dir property holds the value as a path to name node describing where dfs should store its name table (fsimage) on local file system

dfs.datanode.name.dir property determines where the dfs datanode should store its blocks on the local file system

To make hdfs and hadoop commands accessible in terminal we need to setup hadoop path in PATH variable

echo 'export PATH=$PATH:/home/geek/hadoop-2.9.2/bin' >> ~/.bashrc

echo 'export PATH=$PATH:/home/geek/hadoop-2.9.2/sbin' >> ~/.bashrc

Run following command to apply changes.

source ~/.bashrc

If things have setup right. You will be able to execute the following command. Use new terminal for this commad.

hdfs namenode -format

This command simply format the name node as we have been setting it up for the first time so nothing will be lost from the data note.

## Step 4 – Start Hadoop

Enter the following command and you’ll be able to start name node.

start-dfs.sh

You’ll be asked to use the localhost

Then a password will be prompted for local machine.

Now start hadoop module YARN using the below command

start-yarn.sh

You’ll be prompted with local machine password and the instance will be up and running as a daemon.

Lets check our installation using Hdfs command.

hdfs dfs -mkdir /temp-folder

hdfs dfs -ls /

Following result will be shown

drwxr-xr-x - geek supergroup 0 2020-06-13 00:55 /temp-folder

You’ll able to run command on the terminal and can access the GUI of cluster using the following link

http://localhost:50070/

http://localhost:8088/cluster

If you are able to see above pages. Congratulation you have configured Hadoop