**3. Recommended Platform:**

* OS: Linux is supported as a development and production platform. You can use Ubuntu 14.04 or later (you can also use other Linux flavors like: CentOS, Redhat, etc.)
* Hadoop: Cloudera Distribution for Apache hadoop CDH5.x (you can use Apache hadoop 2.x)

**3.1 Setup Platform**

If you are using Windows/Mac OS you can [create virtual machine and install Ubuntu using VMWare Player](http://data-flair.training/blogs/install-ubuntu-vmware-player/), alternatively you can [create virtual machine and install Ubuntu using Oracle Virtual Box](http://data-flair.training/blogs/step-by-step-installation-of-ubuntu-on-virtual-box/).

**4. Prerequisites:**

**4.1. Install Java 7 (Recommended Oracle Java)**

**4.1.1. Install Python Software Properties**

|  |  |
| --- | --- |
| 1 | $sudo apt-get install python-software-properties |

**4.1.2. Add Repository**

|  |  |
| --- | --- |
| 1 | $sudo add-apt-repository ppa:webupd8team/java |

**4.1.3. Update the source list**

|  |  |
| --- | --- |
| 1 | $sudo apt-get update |

**4.1.4. Install Java**

|  |  |
| --- | --- |
| 1 | $sudo apt-get install oracle-java7-installer |

**4.2. Configure SSH**

**4.2.1. Install Open SSH Server-Client**

|  |  |
| --- | --- |
| 1 | $sudo apt-get install openssh-server openssh-client |

**4.2.2. Generate Key Pairs**

|  |  |
| --- | --- |
| 1 | $ssh-keygen -t rsa -P "" |

**4.2.3. Configure password-less SSH**

|  |  |
| --- | --- |
| 1 | $cat $HOME/.ssh/id\_rsa.pub >> $HOME/.ssh/authorized\_keys |

**4.2.4. Check by SSH to localhost**

|  |  |
| --- | --- |
| 1 | $ssh localhost |

**5. Install Hadoop**

**5.1. Download Hadoop**

<http://archive.cloudera.com/cdh5/cdh/5/hadoop-2.5.0-cdh5.3.2.tar.gz>

**5.2. Untar Tar ball**

|  |  |
| --- | --- |
| 1 | $tar xzf hadoop-2.5.0-cdh5.3.2.tar.gz |

*Note: All the required jars, scripts, configuration files, etc. are available in HADOOP\_HOME directory (hadoop-2.5.0-cdh5.3.2)*

**5.3. Setup Configuration:**

**5.3.1. Edit .bashrc:**

Edit .bashrc file located in user’s home directory and add following parameters:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | export HADOOP\_PREFIX="/home/hdadmin/hadoop-2.5.0-cdh5.3.2"  export PATH=$PATH:$HADOOP\_PREFIX/bin  export PATH=$PATH:$HADOOP\_PREFIX/sbin  export HADOOP\_MAPRED\_HOME=${HADOOP\_PREFIX}  export HADOOP\_COMMON\_HOME=${HADOOP\_PREFIX}  export HADOOP\_HDFS\_HOME=${HADOOP\_PREFIX}  export YARN\_HOME=${HADOOP\_PREFIX} |

*Note: After above step restart the terminal, so that all the environment variables will come into effect*

**5.3.2. Edit hadoop-env.sh:**

Edit configuration file hadoop-env.sh (located in HADOOP\_HOME/etc/hadoop) and set JAVA\_HOME:

|  |  |
| --- | --- |
| 1 | export JAVA\_HOME=<path-to-the-root-of-your-Java-installation> (eg: /usr/lib/jvm/java-7-oracle/) |

**5.3.3. Edit core-site.xml:**

Edit configuration file core-site.xml (located in HADOOP\_HOME/etc/hadoop) and add following entries:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | <configuration>      <property>          <name>fs.defaultFS</name>          <value>hdfs://localhost:9000</value>      </property>      <property>          <name>hadoop.tmp.dir</name>          <value>/home/dataflair/hdata</value>      </property>  </configuration> |

*Note: /home/hdadmin/hdata is a sample location; please specify a location where you have Read Write privileges*

**5.3.4. Edit hdfs-site.xml:**

Edit configuration file hdfs-site.xml (located in HADOOP\_HOME/etc/hadoop) and add following entries:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | <configuration>      <property>          <name>dfs.replication</name>          <value>1</value>      </property>  </configuration> |

**5.3.5. Edit mapred-site.xml:**

Edit configuration file mapred-site.xml (located in HADOOP\_HOME/etc/hadoop) and add following entries:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | <configuration>      <property>          <name>mapreduce.framework.name</name>          <value>yarn</value>      </property>  </configuration> |

**5.3.6. Edit yarn-site.xml:**

Edit configuration file mapred-site.xml (located in HADOOP\_HOME/etc/hadoop) and add following entries:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | <configuration>      <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>  </configuration> |

**6. Start the Cluster:**

**6.1. Format the name node:**

|  |  |
| --- | --- |
| 1 | $bin/hdfs namenode -format |

*NOTE: This activity should be done once when you install hadoop, else It will delete all your data from HDFS*

**6.2. Start HDFS Services:**

|  |  |
| --- | --- |
| 1 | $sbin/start-dfs.sh |

**6.3. Start YARN Services:**

|  |  |
| --- | --- |
| 1 | $sbin/start-yarn.sh |

**6.4. Check whether services have been started**

|  |  |
| --- | --- |
| 1  2  3  4  5 | $jps  NameNode  DataNode  ResourceManager  NodeManager |

**7. Run Map-Reduce Jobs**

**7.1. Run word count example:**

|  |  |
| --- | --- |
| 1  2  3  4 | $ bin/hdfs dfs -mkdir /inputwords  $ bin/hdfs dfs -put <data-file> /inputwords  $ bin/yarn jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.5.0-cdh5.3.2.jar wordcount /inputwords /outputwords  $ bin/hdfs dfs -cat /outputwords/\* |

Play with HDFS Commands and perform various operations, [Follow HDFS command Guide](http://data-flair.training/blogs/most-used-hdfs-commands-examples/)

**8. Stop The Cluster**

**8.1. Stop HDFS Services:**

|  |  |
| --- | --- |
| 1 | $sbin/stop-dfs.sh |

**8.2. Stop YARN Services:**

|  |  |
| --- | --- |
| 1 | $sbin/stop-yarn.sh |