**Step by step guide on setting up a Hadoop on an EC2 instance**

After the EC2 is launched, please add the existing ssh-key to localhost, to enable password-less login

ssh-copy-id <username>@localhost

Note: When an EC2 instance is launched, there is a preset ssh-key located in ~/.ssh/, which is the default location and does not need to be specified in the command above.

**Install Hadoop [1]**

**Step 1:**[Click here](https://goo.gl/ipdJJa) to download the JDK 8 Package. Save this file in your home directory.

**Step 2:** Extract the Java Tar File.

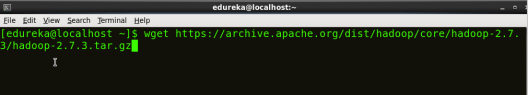
***Command*:** tar -xvf jdk-8u101-linux-i586.tar.gz

Untar Java - Install Hadoop - Edureka

*Fig: Hadoop Installation – Extracting Java Files*

**Step 3:**Download the Hadoop 2.7.3 Package.

***Command*:** wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.3/hadoop-2.7.3.tar.gz



*Fig: Hadoop Installation – Downloading Hadoop*

**Step 4:**Extract the Hadoop tar File.

***Command***: tar -xvf hadoop-2.7.3.tar.gz

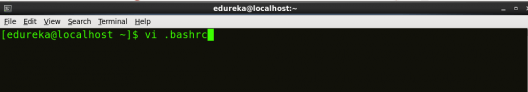
Extract Hadoop Package - Install Hadoop - Edureka

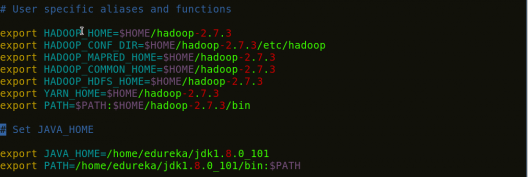
*Fig: Hadoop Installation – Extracting Hadoop Files*

**Step 5:**Add the Hadoop and Java paths in the bash file (.bashrc).

Open ~/**.bashrc** file. Now, add Hadoop and Java Path as shown below.

***Command*:**  vi ~/.bashrc



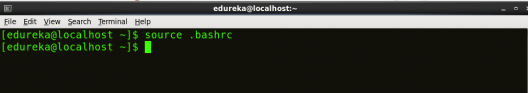


*Fig: Hadoop Installation – Setting Environment Variable*

Then, save the bash file and close it.

For applying all these changes to the current Terminal, execute the source command.

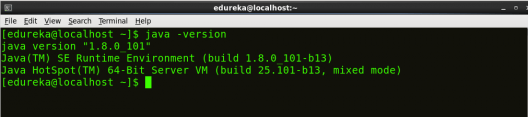
***Command*:** source ~/.bashrc



*Fig: Hadoop Installation – Refreshing environment variables*

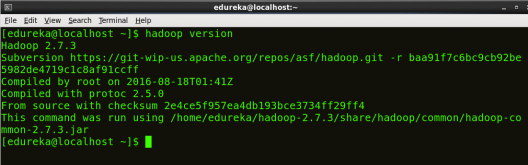
To make sure that Java and Hadoop have been properly installed on your system and can be accessed through the Terminal, execute the java -version and hadoop version commands.

***Command*:**java -version



*Fig: Hadoop Installation – Checking Java Version*

***Command*:**hadoop version



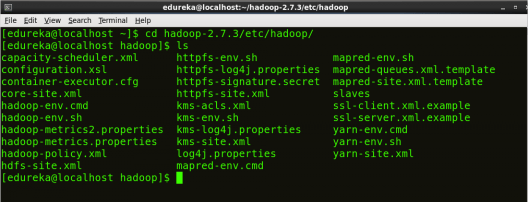
*Fig: Hadoop Installation – Checking Hadoop Version*

**Step 6:**Edit the [**Hadoop Configuration files**](https://www.edureka.co/blog/explaining-hadoop-configuration/).

***Command:*** cd hadoop-2.7.3/etc/hadoop/

***Command:*** ls

All the Hadoop configuration files are located in **hadoop-2.7.3/etc/hadoop** directory as you can see in the snapshot below:

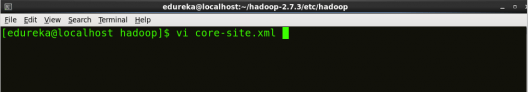


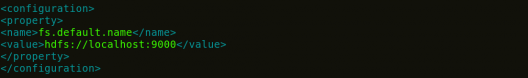
*Fig: Hadoop Installation – Hadoop Configuration Files*

**Step 7:**Open *core-site.xml* and edit the property mentioned below inside configuration tag:

*core-site.xml* informs Hadoop daemon where NameNode runs in the cluster. It contains configuration settings of Hadoop core such as I/O settings that are common to HDFS & MapReduce.

***Command*:** vi core-site.xml





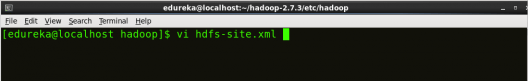
*Fig: Hadoop Installation – Configuring core-site.xml*

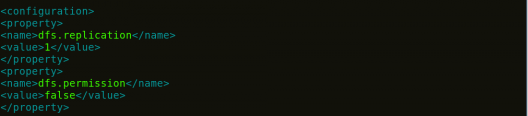
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>fs.default.name</name>  <value>hdfs://localhost:9000</value>  </property>  </configuration> |

**Step 8:** Edit *hdfs-site.xml*and edit the property mentioned below inside configuration tag:

*hdfs-site.xml* contains configuration settings of HDFS daemons (i.e. NameNode, DataNode, Secondary NameNode). It also includes the replication factor and block size of HDFS.

***Command*:** vi hdfs-site.xml





*Fig: Hadoop Installation – Configuring hdfs-site.xml*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | <?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.permission</name>  <value>false</value>  </property>  </configuration> |

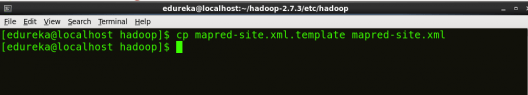
**Step 9:**Edit the *mapred-site.xml* file and edit the property mentioned below inside configuration tag:

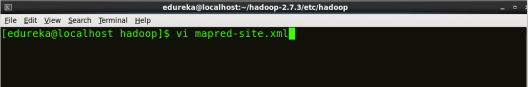
*mapred-site.xml* contains configuration settings of MapReduce application like number of JVM that can run in parallel, the size of the mapper and the reducer process,  CPU cores available for a process, etc.

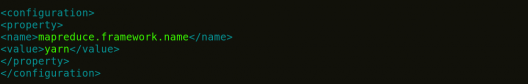
In some cases, mapred-site.xml file is not available. So, we have to create the mapred-site.xml file using mapred-site.xml template.

***Command*:** cp mapred-site.xml.template mapred-site.xml

***Command*:**vi mapred-site.xml.







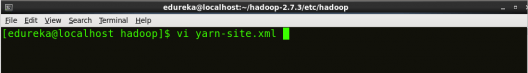
*Fig: Hadoop Installation – Configuring mapred-site.xml*

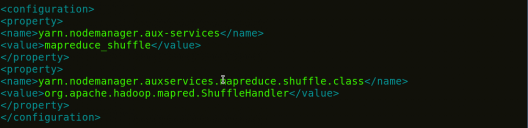
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  </configuration> |

**Step 10:** Edit *yarn-site.xml* and edit the property mentioned below inside configuration tag:

*yarn-site.xml* contains configuration settings of ResourceManager and NodeManager like application memory management size, the operation needed on program & algorithm, etc.

***Command*:** vi yarn-site.xml





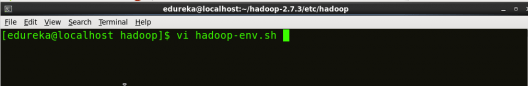
*Fig: Hadoop Installation – Configuring yarn-site.xml*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | <?xml version="1.0">  <configuration>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <property>  <name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>  <value>org.apache.hadoop.mapred.ShuffleHandler</value>  </property>  </configuration> |

**Step 11:**Edit*hadoop-env.sh* and add the Java Path as mentioned below:

*hadoop-env.sh* contains the environment variables that are used in the script to run Hadoop like Java home path, etc.

***Command*:** vi hadoop–env.sh



Property of hadoop-env - Install Hadoop - Edureka

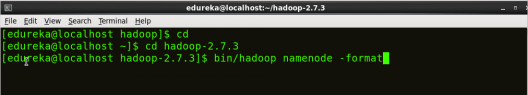
*Fig: Hadoop Installation – Configuring hadoop-env.sh*

**Step 12:** Go to Hadoop home directory and format the NameNode.

***Command*:** cd

***Command*:** cd hadoop-2.7.3

***Command*:** bin/hadoop namenode -format



*Fig: Hadoop Installation – Formatting NameNode*

This formats the HDFS via NameNode. This command is only executed for the first time. Formatting the file system means initializing the directory specified by the dfs.name.dir variable.

Never format, up and running Hadoop filesystem. You will lose all your data stored in the HDFS.

**Step 13:** Once the NameNode is formatted, go to hadoop-2.7.3/sbin directory and start all the daemons.

***Command:***cd hadoop-2.7.3/sbin

Either you can start all daemons with a single command or do it individually.

***Command:****./*start-all.sh

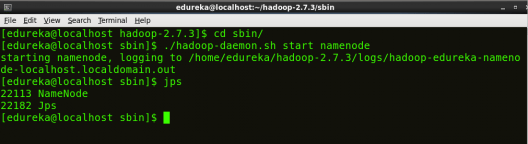
The above command is a combination of***start-dfs.sh, start-yarn.sh*** & ***mr-jobhistory-daemon.sh***

Or you can run all the services individually as below:

**Start NameNode:**

The NameNode is the centerpiece of an HDFS file system. It keeps the directory tree of all files stored in the HDFS and tracks all the file stored across the cluster.

***Command:***./hadoop-daemon.sh start namenode

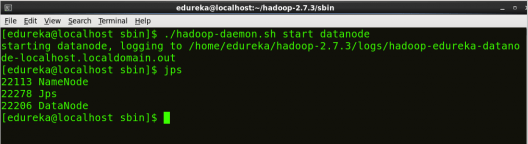


*Fig: Hadoop Installation – Starting NameNode*

**Start DataNode:**

On startup, a DataNode connects to the Namenode and it responds to the requests from the Namenode for different operations.

***Command:***./hadoop-daemon.sh start datanode

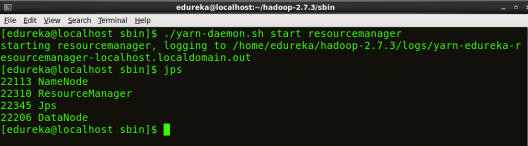


*Fig: Hadoop Installation – Starting DataNode*

**Start ResourceManager:**

ResourceManager is the master that arbitrates all the available cluster resources and thus helps in managing the distributed applications running on the YARN system. Its work is to manage each NodeManagers and the each application’s ApplicationMaster.

***Command:***./yarn-daemon.sh start resourcemanager

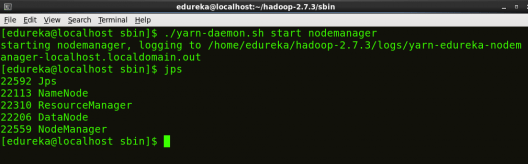


*Fig: Hadoop Installation – Starting ResourceManager*

**Start NodeManager:**

The NodeManager in each machine framework is the agent which is responsible for managing containers, monitoring their resource usage and reporting the same to the ResourceManager.

***Command:***./yarn-daemon.sh start nodemanager



*Fig: Hadoop Installation – Starting NodeManager*

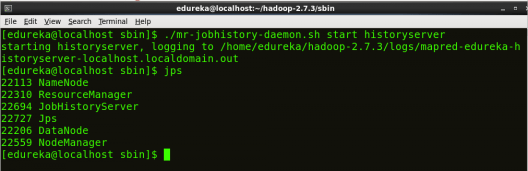
**Start JobHistoryServer:**

JobHistoryServer is responsible for servicing all job history related requests from client.

***Command*:** ./mr-jobhistory-daemon.sh start historyserver

**Step 14:** To check that all the Hadoop services are up and running, run the below command.

***Command:***jps



*Fig: Hadoop Installation – Checking Daemons*

To this end, the Hadoop should be successfully configured and launched.

Reference:

[1] Install Hadoop: Setting up a Single Node Hadoop Cluster. https://www.edureka.co/blog/install-hadoop-single-node-hadoop-cluster