How to Install Hadoop on Mac with Homebrew

Elaborated by Yudith Cardinale

Step 1: Install Java

Open the terminal and enter *java -version* to check the current Java version. If no version is returned then go for the [official website]() () to install it

After the installation is complete, configure the environment variables of JAVA. JDK is installed in the directory */Library/Java/JavaVirtualMachines*

```
ls /Library/Java/JavaVirtualMachines
adoptopenjdk-8.jdk jdk1.8.0_291.jdk
```

Enter *vim ~/.bash_profile or vim ~/.zprofile* in the terminal to configure the Java path.

Place the following statement in the blank line

```
export JAVA_HOME=/Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk/Contents/Home
```

Then, execute *source ~/.bash_profile or source ~/.zprofile* in the terminal to make the configuration file effective.

```
source ~/.zprofile
```

Then enter the *java -version* in the terminal, you can see the Java version.

```
yudithcardinale@MacBookYudith% java -version
openjdk version "1.8.0_292"
OpenJDK Runtime Environment (AdoptOpenJDK)(build 1.8.0_292-b10)
OpenJDK 64-Bit Server VM (AdoptOpenJDK)(build 25.292-b10, mixed mode)
```

Step 2: Install Homebrew

Homebrew is very commonly used on mac, not much to describe, installation method, it is just to execute the following command:

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Enter *vim ~/.bash_profile or vim ~/.zprofile* in the terminal to add the following statement:

```
eval "$(/opt/homebrew/bin/brew shellenv)"
```

Step 3: Configure ssh passwordless

```
ssh-keygen -t rsa -P'' -f ~/.ssh/id_rsa
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

After this step open terminal and enter "ssh localhost", you should log in without a password and that indicates your settings is successful

```
ssh localhost
```

You will be asked to authenticate hosts by adding RSA keys to known hosts. Type yes and hit Enter to authenticate the localhost. Answer yes

Step 4: Install Hadoop

Execute the following command in the terminal:
```
brew install hadoop
```

After installation is completed verify hadoop path

```
yudithcardinale@MacBookYudith% cd /opt/homebrew/Cellar/hadoop/3.3.3/
yudithcardinale@MacBook-Pro-de-Yudith 3.3.3 % ls
INSTALL_RECEIPT.json    NOTICE.txt          libexec
LICENSE-binary          README.txt          sbin
LICENSE.txt             bin
```

Note: latest hadoop will be installed. In this case Hadoop 3.3.3 at the time of writing this document. After the installation is complete, enter "hadoop version" to view the version. If there is an information receipt, the installation is successful.

```
yudithcardinale@MacBookYudith ~ % hadoop version

Hadoop 3.3.3
Source code repository https://github.com/apache/hadoop.git -r
d37586cbda38c338d9fe481addda5a05fb516f71
Compiled by stevel on 2022-05-09T16:36Z
Compiled with protoc 3.7.1
From source with checksum eb96dd4a797b6989ae0cdb9db6efc6

This command was run using

/opt/homebrew/Cellar/hadoop/3.3.3/libexec/share/hadoop/common/hadoop-common-

3.3.3.jar
```

Step 5: Hadoop configuration

Make sure that all environment variables are wel set. Your .bash_profile or .zprofile looks like:

```
eval "$(/opt/homebrew/bin/brew shellenv)"
export JAVA_HOME=/Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk/Contents/Home


export PATH=$PATH:$JAVA_HOME/bin

## HADOOP env variables
export HADOOP_HOME="/opt/homebrew/Cellar/hadoop/3.3.3/libexec"
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export HADOOP_MAPRED_HOME=$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 HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar
```

Until this step, you have the standalone configuration of hadoop

Step 6: Pseudo-distributed hadoop configuration

For the pseudo-distributed operating mode, we need to configure the following files:

Core-site.xml

Open $HADOOP_HOME/etc/hadoop/Core-site.xml in terminal and add below properties

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

Hdfs-site.xml

Open $HADOOP_HOME/etc/hadoop/Hdfs-site.xml file in terminal and add below properties

```
<configuration>
   <property>
```

```
      <name>dfs.replication</name>
      <value>1</value>
   </property>
</configuration>
```

Yarn-site.xml

Open $HADOOP_HOME/etc/hadoop/yarn-site.xml file and add below properties

```
<configuration>
<!-- Site specific YARN configuration properties →
   <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_PRE
PEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_MAPRED_HOME</value>
   </property>
</configuration>
```

Mapred-site.xml

Open $HADOOP_HOME/etc/hadoop/mapred-site.xml file in termial and add below properties.

```
<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_HOME/share/
hadoop/mapreduce/lib/*</value>
  </property>
</configuration>
```

hdfs format

```
$HADOOP_HOME/bin/hdfs namenode -format
```

Note: Open terminal and Initialize Hadoop cluster by formatting HDFS directory

Step 7: Final Step

Run start-all.sh in the sbin folder

```
$HADOOP_HOME/sbin/start-all.sh

Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [K9-MAC-061.local]
Starting resourcemanager
Starting nodemanagers
```

Use JPS command to check if all name node, Data node, resource manager is started successfully

```
4929 DataNode
5294 NodeManager
5200 ResourceManager
5354 Jps
5046 SecondaryNameNode
4831 NameNode
```

How to Access Hadoop web interfaces (Hadoop Health)

```
NameNode                 : http://localhost:9870
NodeManager              : http://localhost:8042
Resource Manager (Yarn)  : http://localhost:8088/cluster
```