

Installing Hadoop on the Windows Subsystem for Linux

Simon A. Broda
simon.broda@uzh.ch

October 11, 2018

To run [Hadoop](#) under Microsoft Windows, you can either [build from source](#) or [follow this guide](#) to use the windows binaries (not tested). On Windows 10, a third option is to use the Windows Subsystem for Linux (WSL). To do this, install WSL following the instructions [here](#). Below, it is assumed that you chose Ubuntu; the commands will be different for other distributions.

- The first step is to install Java in Ubuntu; to do that, open bash from the start menu (or from search). Bash is the *shell* for Ubuntu on WSL. In it, you do

```
sudo apt-get install default-jdk
```

- After that, you can follow [this guide](#): start by doing

```
sudo apt purge openssh-server  
sudo apt install openssh-server  
sudo service ssh start
```

to fix an issue with SSH.

- Then, do

```
update-alternatives --config java
```

and write down the path that it gives you, up to but not including the `jre/bin/java` at the end. For me, this gives `/usr/lib/jvm/java-8-openjdk-amd64`.

- We will set up Hadoop for pseudo-distributed operation, following [this guide](#): start by installing some required packages by doing

```
sudo apt-get install ssh  
sudo apt-get install rsync
```

- Then, download Hadoop:

```
wget http://www-eu.apache.org/dist/hadoop/common/  
hadoop-2.9.1/hadoop-2.9.1.tar.gz
```

(the backslash is a line continuation character in bash; this is so the command fits on this page. You can also delete the backslash and put the whole command on one line).

- Unpack it:

```
tar -xvzf hadoop-2.9.1.tar.gz
```

- Next, you do

```
cd hadoop-2.9.1
nano etc/hadoop/hadoop-env.sh
```

This opens an editor (nano). In nano, using `ctrl-o` to save, `ctrl-x` to exit, change the line

```
export JAVA_HOME=${JAVA_HOME}
```

so that the path equals the one you wrote down earlier. Be careful to put a slash and no space before the path; for me, this yields

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

- It remains to configure Hadoop for pseudo-distributed operation. Again using nano,

```
nano etc/hadoop/core-site.xml
```

In that file, change

```
<configuration>

</configuration>
```

to

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

(you should be able to paste into the bash window by just right-clicking).

- Finally, do

```
nano etc/hadoop/hdfs-site.xml
```

and modify

```
<configuration>

</configuration>
```

to

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

- That should pretty much be it. You should now be able to do

```
bin/hdfs namenode -format
sbin/start-dfs.sh
```

and then point your browser at `http://localhost:50070`. Note that while Hadoop starts up, you may be prompted several times to confirm whether you are sure that you want to connect; just enter `yes<enter>`. The default is no, so don't just hit `<enter>`. You may also need to enter your password several times. You can ignore the warning about not being able to set the niceness.

- To shut down Hadoop, do

```
sbin/stop-dfs.sh
```

- Note that the next time you open bash, you have to do

```
cd hadoop-2.9.1
```

for the above commands to work again without modification.