How to Install Hadoop in Stand-Alone Mode on Ubuntu 18.04

Big data is data that contains a greater variety arriving in increasing volumes and with ever-higher velocity. This is known as the three Vs.

What is Hadoop?

Hadoop is a platform for distributed computing and storage across a number of servers. Hadoop is designed for batch processing.
First developed and released as open source by Yahoo at 2007. It implements the MapReduce approach to process data. Hadoop's MapReduce involves distributing a dataset among multiple servers and operating on the data which is the Map stage. The partial results are then recombined which known as Reduce stage.

To store data Hadoop utilizes its own distributed filesystem, HDFS which make data available to multiple computing nodes.

Hadoop Installations:

Hadoop can run in three different mode as below :

Local installation :

•All daemons run in a single process (single JVM).
•Very fast, but local processing.
•Apps tested in this mode are not guaranteed to work on the cluster.

Psuedo Distribution (Single Node Cluster):

•All daemons run in single machine but different processes (multiple JVMs).
•Typical dev machine setup.

Full Distribution (Multi Node Cluster):

•All daemons run in multiples machines in the network [in different processes (multiple JVMs)].
•Typical production/test cluster setup.

In this post we will able to execute hadoop stand-alone installation .

1. Prepare machine.
   Install java-1.8-64 bit & ssh on machine.

2. Disable ipv6 on all machines -- /etc/sysctl.conf
   net.ipv6.conf.all.disable_ipv6 = 1
   net.ipv6.conf.default.disable_ipv6 = 1
   net.ipv6.conf.lo.disable_ipv6 = 1

3. In /etc/hosts ensure entry of standalone hostname.
   127.0.0.1   localhost

4. Enable password-less login for SSH:
```
ssh-keygen -t rsa -P ""
cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
chmod 600 $HOME/.ssh/authorized_keys
ssh localhost
```

5. Download & Extract into $HOME Hadoop 2.7.3.

```
cd ~
wget https://archive.apache.org/dist/hadoop/core/hadoop-2.7.3/hadoop-2.7.3.tar.gz
tar xvf hadoop-2.7.3.tar.gz
```

6. In $HOME/.bashrc setup HADOOP_HOME and PATH:

```
export HADOOP_HOME=$HOME/ hadoop-2.7.3
export PATH=$HADOOP_HOME/bin:$HADOOP_HOME/sbin:$PATH
```

7. In $HADOOP_HOME/etc/hadoop/hadoop-env.sh:

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

8. In $HADOOP_HOME/etc/hadoop/core-site.xml:

```
<?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>
```

9. In $HADOOP_HOME/etc/hadoop/hdfs-site.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
      <property>
      <name>dfs.name.dir</name>
      <value>${user.home}/bigdata/hd-data/nn</value>
      </property>
      <property>
      <name>dfs.replication</name>
      <value>1</value>
      </property>
      <property>
      <name>dfs.data.dir</name>
      <value>${user.home}/bigdata/hd-data/dn</value>
      </property>
</configuration>
```

10. In $HADOOP_HOME/etc/hadoop/mapred-site.xml:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
      <property>
      <name>mapreduce.framework.name</name>
      <value>yarn</value>
      </property>
</configuration>
```

11. In $HADOOP_HOME/etc/hadoop/yarn-site.xml:

```
<?xml version="1.0"?>
<configuration>
      <property>
      <name>yarn.resourcemanager.hostname</name>
      <value>localhost</value>
      </property>
      <property>
```

```
            <name>yarn.nodemanager.aux-services</name>
            <value>mapreduce_shuffle</value>
            </property>
            <property>
            <name>yarn.nodemanager.local-dirs</name>
            <value>${user.home}/bigdata/hd-data/yarn/data</value>
            </property>
            <property>
            <name>yarn.nodemanager.logs-dirs</name>
            <value>${user.home}/bigdata/hd-data/yarn/logs</value>
            </property>
            <property>
                  <name>yarn.nodemanager.disk-health-checker.max-disk-
utilization-per-disk-percentage</name>
                  <value>99.9</value>
            </property>
      </configuration>

12.
       $HADOOP_HOME/etc/hadoop/slaves
            localhost

13. Format namenode:
            hdfs namenode -format

14. Start HDFS & YARN:
            start-dfs.sh
            start-yarn.sh
      Verify daemons using "jps" command.

15. HDFS commands:
            hadoop fs -ls /
            hadoop fs -mkdir /user/data
            hadoop fs -put localfilepath /user/data
            hadoop fs -get /user/data/filepath localfilepath

16. Stop HDFS & YARN:
            stop-yarn.sh
            stop-dfs.sh
      Verify daemons using "jps" command.
```