

Apache Hadoop 3.3.6 Installation on Ubuntu 22.04

Step 1 : Install Java Development Kit

The default Ubuntu repositories contain Java 8 and Java 11 both. I am using Java 8 because hive only works on this version. Use the following command to install it.

```
sudo apt update && sudo apt install openjdk-8-jdk
```

Step 2 : Verify the Java version :

Once you have successfully installed it, check the current Java version:

```
java -version
```

A terminal window screenshot showing the output of the 'java -version' command. The output displays the OpenJDK version (11.0.20.1), the build date (2023-08-24), the runtime environment details (build 11.0.20.1+1-post-Ubuntu-0ubuntu123.04), and the server VM details (build 11.0.20.1+1-post-Ubuntu-0ubuntu123.04, mixed mode, sharing). The prompt is 'sanjay@sanjay-VirtualBox:~\$' and the cursor is at the end of the line.

```
sanjay@sanjay-VirtualBox:~$ java -version
openjdk 11.0.20.1 2023-08-24
OpenJDK Runtime Environment (build 11.0.20.1+1-post-Ubuntu-0ubuntu123.04)
OpenJDK 64-Bit Server VM (build 11.0.20.1+1-post-Ubuntu-0ubuntu123.04, mixed mode, sharing)
sanjay@sanjay-VirtualBox:~$
```

Step 3 : Install SSH :

SSH (Secure Shell) installation is vital for Hadoop as it enables secure communication between nodes in the Hadoop cluster. This ensures

data integrity, confidentiality, and allows for efficient distributed processing of data across the cluster.

```
sudo apt install ssh
```

Step 4 : Create the hadoop user :

All the Hadoop components will run as the user that you create for Apache Hadoop, and the user will also be used for logging in to Hadoop's web interface.

Run the command to create user and set password :

```
sudo adduser hadoop
```

```
sanjay@sanjay-VirtualBox:~$ sudo adduser hadoop
Adding user 'hadoop' ...
Adding new group 'hadoop' (1001) ...
Adding new user 'hadoop' (1001) with group 'hadoop (1001)' ...
adduser: The home directory '/home/hadoop' already exists. Not copying from '/etc/skel'.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: password updated successfully
Changing the user information for hadoop
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
Adding new user 'hadoop' to supplemental / extra groups 'users' ...
Adding user 'hadoop' to group 'users' ...
sanjay@sanjay-VirtualBox:~$ su - hadoop
Password:
hadoop@sanjay-VirtualBox:~$
```

Step 5 : Switch user :

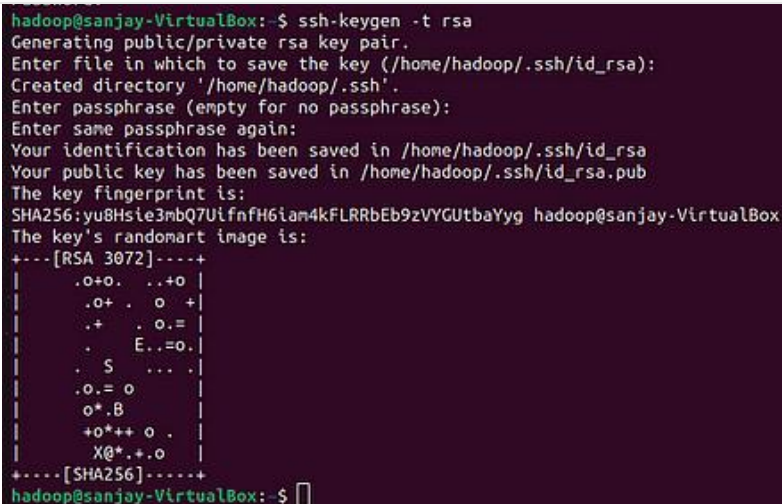
Switch to the newly created hadoop user:

```
su - hadoop
```

Step 6 : Configure SSH :

Now configure password-less SSH access for the newly created hadoop user, so I didn't enter key to save file and passphrase. Generate an SSH keypair first:

```
ssh-keygen -t rsa
```



```
hadoop@sanjay-VirtualBox:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoop/.ssh/id_rsa):
Created directory '/home/hadoop/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoop/.ssh/id_rsa
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:yu8Hsie3mbQ7UifnfH6iam4kFLRRbEb9zVYGUtbaYyg hadoop@sanjay-VirtualBox
The key's randomart image is:
+---[RSA 3072]-----+
|  .o+.  ..+o |
|  .o+ .  o  + |
|  .+   . o.= |
|  .   E..=o. |
|  . S   ...  |
|  .o.= o     |
|  o*.B       |
|  +o*++ o .  |
|  X@*+.o     |
+----[SHA256]-----+
hadoop@sanjay-VirtualBox:~$
```

Step 7 : Set permissions :

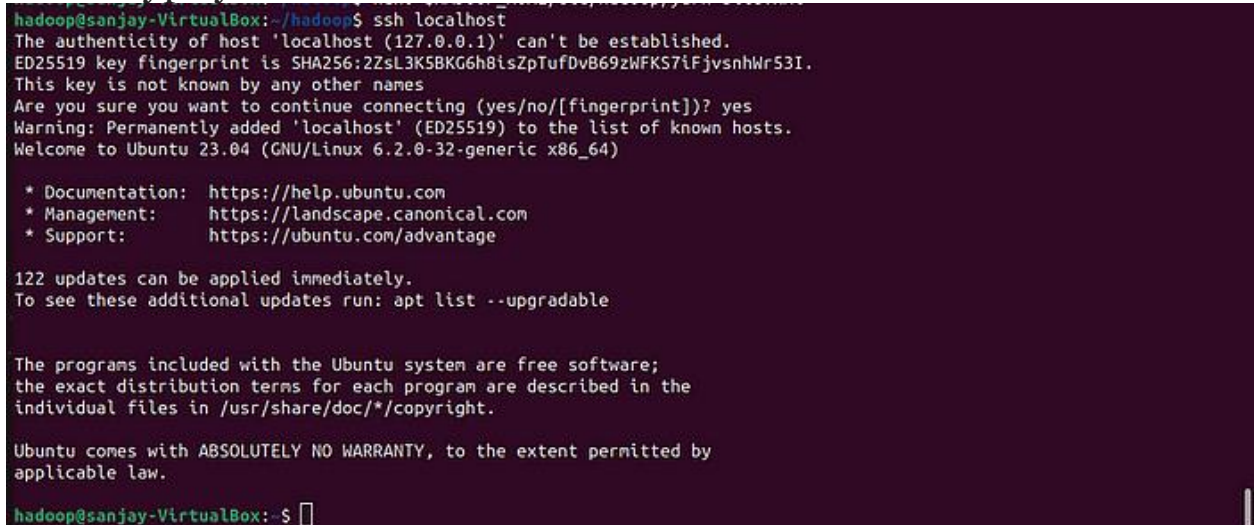
Copy the generated public key to the authorized key file and set the proper permissions:

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
chmod 640 ~/.ssh/authorized_keys
```

Step 8 : SSH to the localhost

```
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.

A terminal window with a dark purple background. The prompt is 'hadoop@sanjay-VirtualBox: ~/hadoop\$'. The user enters 'ssh localhost'. The terminal shows the SSH warning about the host's authenticity, the fingerprint, and the user's confirmation to proceed. It then shows the Ubuntu 23.04 welcome message, update information, and system documentation links.

```
hadoop@sanjay-VirtualBox: ~/hadoop$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:2ZsL3K5BKG6h8isZpTufDvB69zWFKS7iFjvsnhWr53I.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
Welcome to Ubuntu 23.04 (GNU/Linux 6.2.0-32-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

122 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

hadoop@sanjay-VirtualBox: ~$
```

Step 9 : Switch user

Again switch to hadoop

```
su - hadoop
```

Step 10 : Install hadoop

- Download hadoop 3.3.6

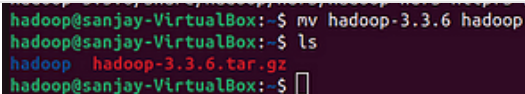
```
wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz
```

- Once you've downloaded the file, you can unzip it to a folder.

```
tar -xvzf hadoop-3.3.6.tar.gz
```

- Rename the extracted folder to remove version information. This is an optional step, but if you don't want to rename, then adjust the remaining configuration paths.

```
mv hadoop-3.3.6 hadoop
```



```
hadoop@sanjay-VirtualBox:~$ mv hadoop-3.3.6 hadoop
hadoop@sanjay-VirtualBox:~$ ls
hadoop  hadoop-3.3.6.tar.gz
hadoop@sanjay-VirtualBox:~$
```

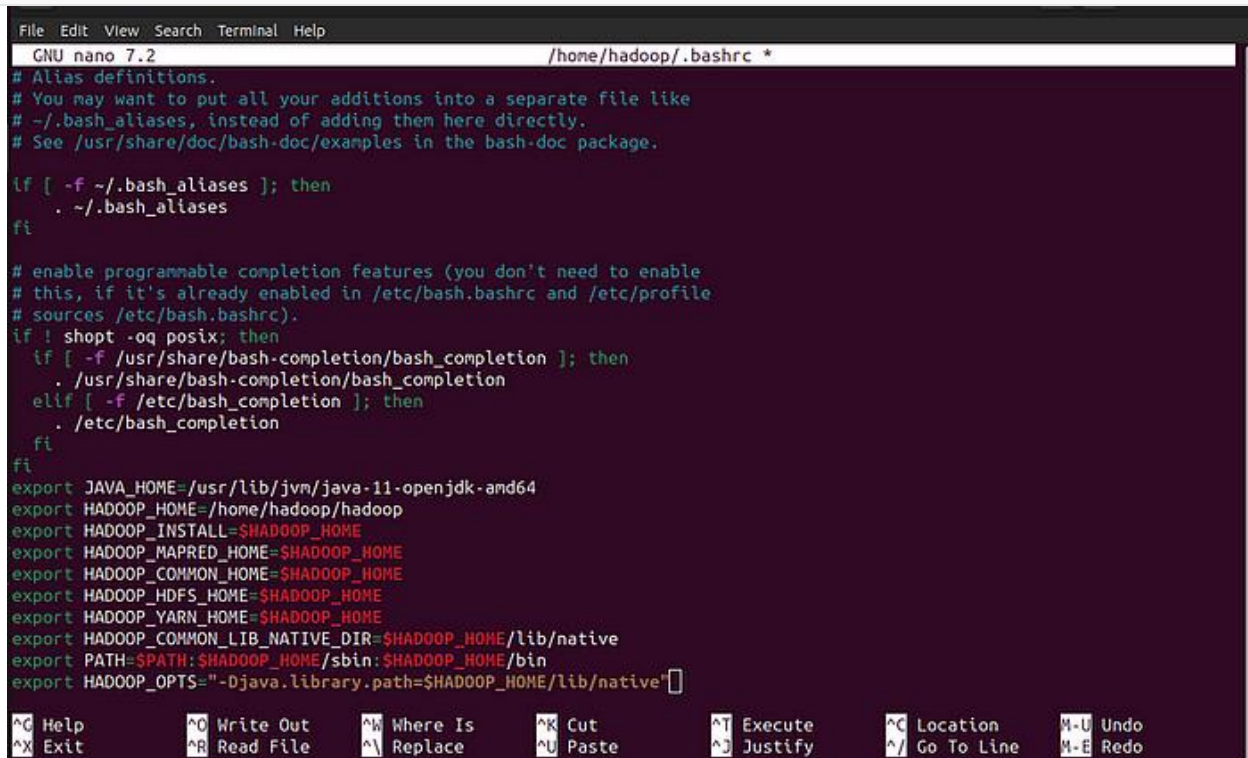
- Next, you will need to configure Hadoop and Java Environment Variables on your system. Open the ~/.bashrc file in your favorite text editor. Here I am using nano editor , to pasting the code we use ctrl+shift+v for saving the file ctrl+x and ctrl+y ,then hit enter:

```
nano ~/.bashrc
```

- Append the below lines to the file.

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export HADOOP_HOME=/home/hadoop/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
```

```
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
```



```
GNU nano 7.2 /home/hadoop/.bashrc *
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export HADOOP_HOME=/home/hadoop/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
```

- Load the above configuration in the current environment.

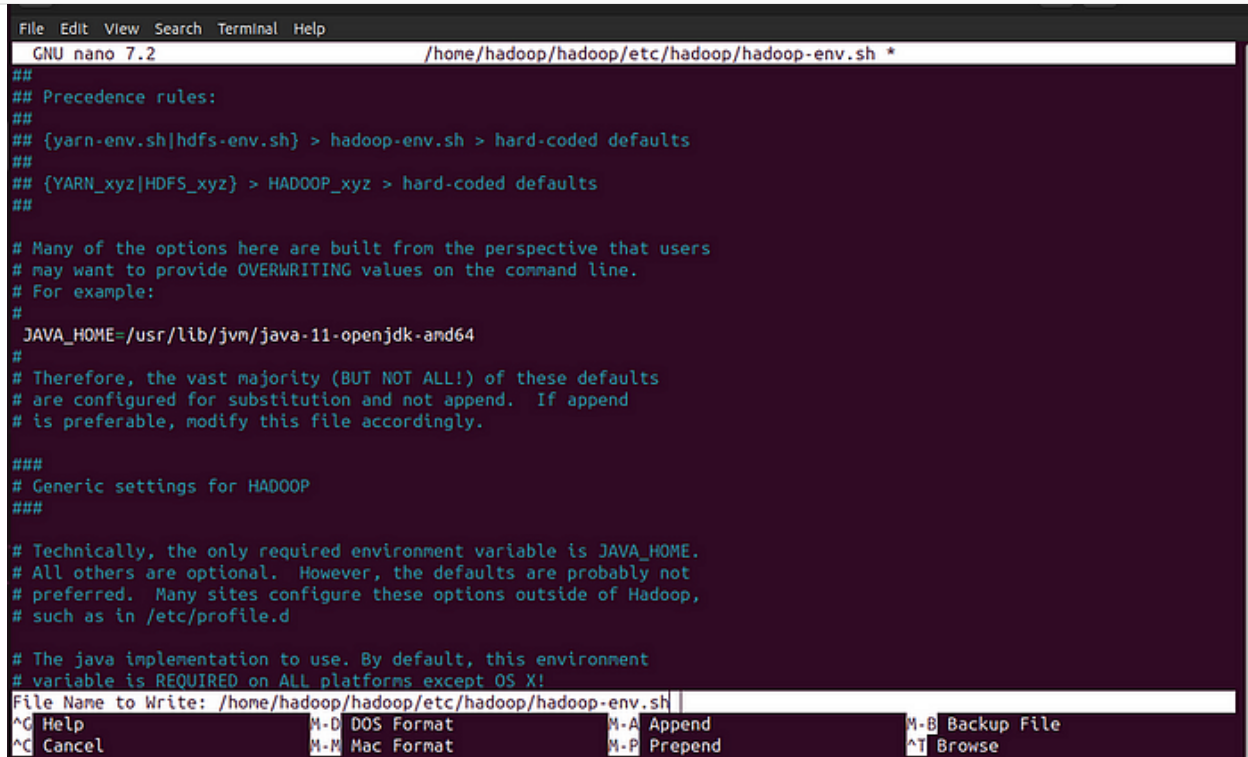
```
source ~/.bashrc
```

- You also need to configure JAVA_HOME in hadoop-env.sh file. Edit the Hadoop environment variable file in the text editor:

```
nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
```

Search for the “export JAVA_HOME” and configure it .

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

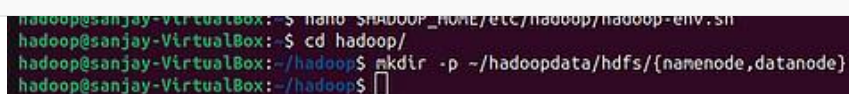


```
File Edit View Search Terminal Help
GNU nano 7.2 /home/hadoop/hadoop/etc/hadoop/hadoop-env.sh *
##
## Precedence rules:
##
## {yarn-env.sh|hdfs-env.sh} > hadoop-env.sh > hard-coded defaults
##
## {YARN_xyz|HDFS_xyz} > HADOOP_xyz > hard-coded defaults
##
# Many of the options here are built from the perspective that users
# may want to provide OVERWRITING values on the command line.
# For example:
#
# JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
#
# Therefore, the vast majority (BUT NOT ALL!) of these defaults
# are configured for substitution and not append. If append
# is preferable, modify this file accordingly.
###
# Generic settings for HADOOP
###
# Technically, the only required environment variable is JAVA_HOME.
# All others are optional. However, the defaults are probably not
# preferred. Many sites configure these options outside of Hadoop,
# such as in /etc/profile.d
#
# The java implementation to use. By default, this environment
# variable is REQUIRED on ALL platforms except OS X!
File Name to Write: /home/hadoop/hadoop/etc/hadoop/hadoop-env.sh
^C Help      M-D DOS Format  M-A Append     M-B Backup File
^C Cancel    M-M Mac Format  M-P Prepend    ^T Browse
```

Step 11 : Configuring Hadoop :

- First, you will need to create the **namenode** and **datanode** directories inside the Hadoop user home directory. Run the following command to create both directories:

```
cd hadoop/
mkdir -p ~/hadoopdata/hdfs/{namenode,datanode}
```



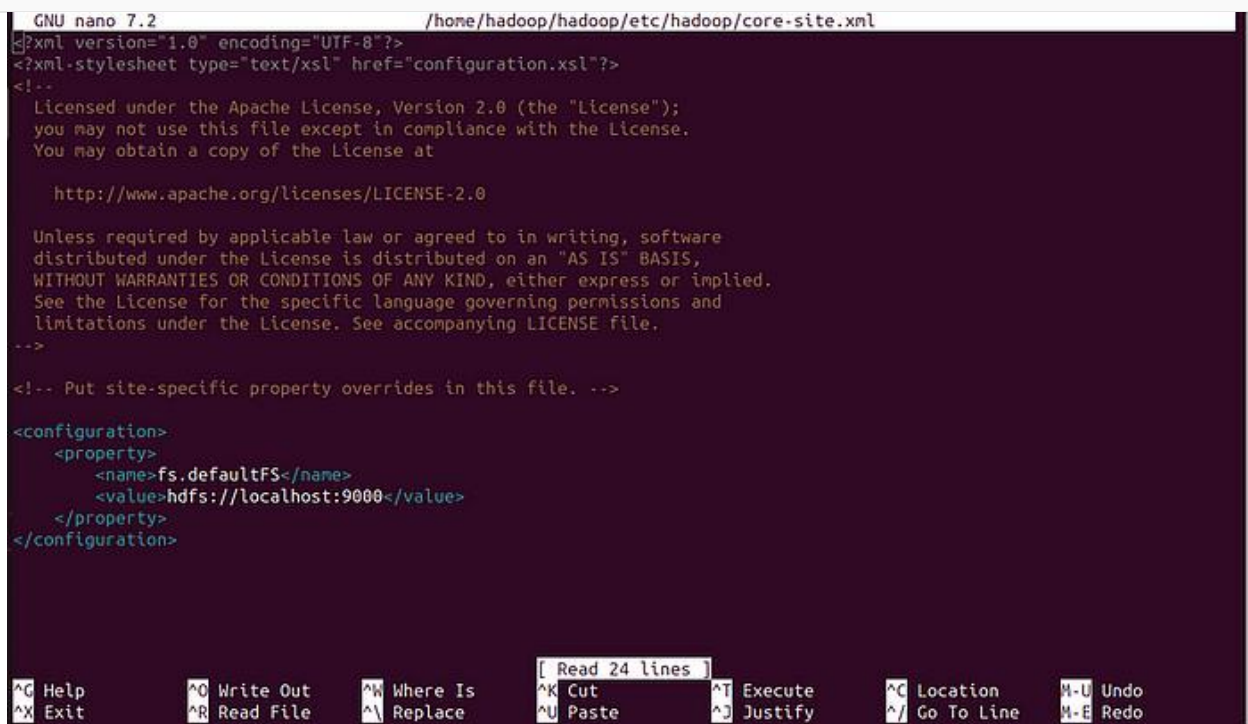
```
hadoop@sanjay-VirtualBox: $ nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
hadoop@sanjay-VirtualBox: $ cd hadoop/
hadoop@sanjay-VirtualBox: ~/hadoop$ mkdir -p ~/hadoopdata/hdfs/{namenode,datanode}
hadoop@sanjay-VirtualBox: ~/hadoop$
```


- Next, edit the core-site.xml file and update with your system hostname:

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

Change the following name as per your system hostname:

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



```
GNU nano 7.2 /home/hadoop/hadoop/etc/hadoop/core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

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

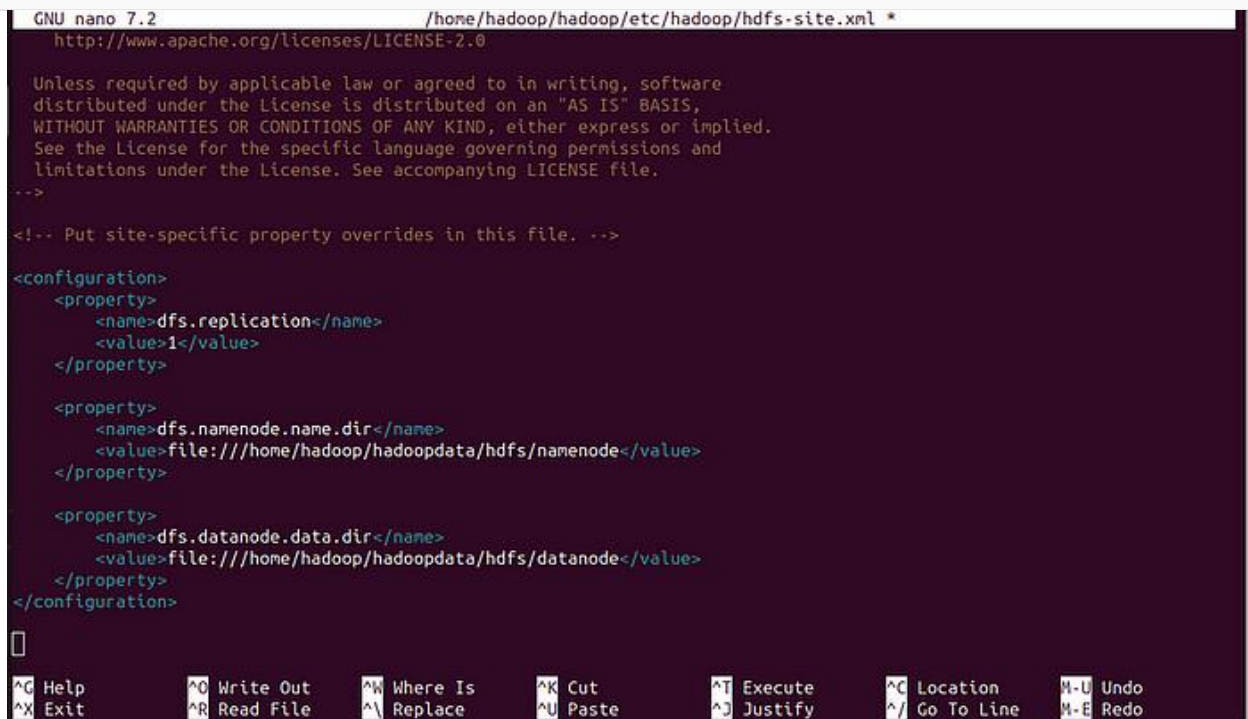
Save and close the file.

- Then, edit the `hdfs-site.xml` file:

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

- Change the NameNode and DataNode directory paths as shown below:

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:///home/hadoop/hadoopdata/hdfs/namenode</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:///home/hadoop/hadoopdata/hdfs/datanode</value>
  </property>
</configuration>
```



```
GNU nano 7.2 /home/hadoop/hadoop/etc/hadoop/hdfs-site.xml *
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the license is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the license for the specific language governing permissions and
limitations under the license. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

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

  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:///home/hadoop/hadoopdata/hdfs/namenode</value>
  </property>

  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:///home/hadoop/hadoopdata/hdfs/datanode</value>
  </property>
</configuration>

[]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line M-E Redo
```

- Then, edit the mapred-site.xml file:

```
nano $HADOOP_HOME/etc/hadoop/mapred-site.xml
```

- Make the following changes:

```
<configuration>
  <property>
    <name>yarn.app.mapreduce.am.env</name>

    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</value>
  </property>
  <property>
    <name>mapreduce.map.env</name>

    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</value>
  </property>
  <property>
    <name>mapreduce.reduce.env</name>

    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</value>
  </property>
</configuration>
```

```
GNU nano 7.2 /home/hadoop/hadoop/etc/hadoop/mapred-site.xml
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>yarn.app.mapreduce.am.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</value>
  </property>
  <property>
    <name>mapreduce.map.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</value>
  </property>
  <property>
    <name>mapreduce.reduce.env</name>
    <value>HADOOP_MAPRED_HOME=$HADOOP_HOME/home/hadoop/hadoop/bin/hadoop</value>
  </property>
</configuration>
```

^O Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo

- Then, edit the yarn-site.xml file:

```
nano $HADOOP_HOME/etc/hadoop/yarn-site.xml
```

- Make the following changes:

```
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
</configuration>
```

```
GNU nano 7.2 /home/hadoop/hadoop/etc/hadoop/yarn-site.xml *
<?xml version="1.0"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
</configuration>[]
```

^G Help ^O Write Out ^W Where Is ^R Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo

Save the file and close it .

Step 12 : Start Hadoop cluster:

- Before starting the Hadoop cluster. You will need to format the **Namenode** as a hadoop user.
- Run the following command to format the Hadoop **Namenode**:

```
hdfs namenode -format
```

- Once the **namenode** directory is successfully formatted with hdfs file system, you will see the message “Storage directory

/home/hadoop/hadoopdata/hdfs/namenode has been successfully formatted”.

```
2023-09-10 13:07:27,704 INFO snapshot.SnapshotManager: Loaded config captureOpenFiles: false, skipCaptureAccessTimeOnly
Change: false, snapshotDiffAllowSnapRootDescendant: true, maxSnapshotLimit: 65536
2023-09-10 13:07:27,712 INFO snapshot.SnapshotManager: SkipList is disabled
2023-09-10 13:07:27,727 INFO util.GSet: Computing capacity for map cachedBlocks
2023-09-10 13:07:27,727 INFO util.GSet: VM type = 64-bit
2023-09-10 13:07:27,727 INFO util.GSet: 0.25% max memory 748 MB = 1.9 MB
2023-09-10 13:07:27,728 INFO util.GSet: capacity = 2^18 = 262144 entries
2023-09-10 13:07:27,744 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
2023-09-10 13:07:27,751 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
2023-09-10 13:07:27,752 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2023-09-10 13:07:27,755 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2023-09-10 13:07:27,756 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expir
y time is 600000 millis
2023-09-10 13:07:27,761 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2023-09-10 13:07:27,761 INFO util.GSet: VM type = 64-bit
2023-09-10 13:07:27,761 INFO util.GSet: 0.029999999329447746% max memory 748 MB = 229.8 KB
2023-09-10 13:07:27,761 INFO util.GSet: capacity = 2^15 = 32768 entries
2023-09-10 13:07:27,804 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1272319295-127.0.1.1-1694331447796
2023-09-10 13:07:27,847 INFO common.Storage: Storage directory /home/hadoop/hadoopdata/hdfs/namenode has been successfu
lly formatted.
2023-09-10 13:07:27,899 INFO namenode.FSImageFormatProtobuf: Saving image file /home/hadoop/hadoopdata/hdfs/namenode/cu
rrent/fsimage.ckpt_000000000000000000 using no compression
2023-09-10 13:07:28,040 INFO namenode.FSImageFormatProtobuf: Image file /home/hadoop/hadoopdata/hdfs/namenode/current/f
simage.ckpt_000000000000000000 of size 401 bytes saved in 0 seconds .
2023-09-10 13:07:28,054 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2023-09-10 13:07:28,074 INFO namenode.FSNamesystem: Stopping services started for active state
2023-09-10 13:07:28,078 INFO namenode.FSNamesystem: Stopping services started for standby state
2023-09-10 13:07:28,087 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2023-09-10 13:07:28,088 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at sanjay-VirtualBox/127.0.1.1
*****/
hadoop@sanjay-VirtualBox:~/hadoop$
```

- Then start the Hadoop cluster with the following command.

```
start-all.sh
```

```
hadoop@sanjay-VirtualBox:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [sanjay-VirtualBox]
Starting resourcemanager
Starting nodemanagers
hadoop@sanjay-VirtualBox:~$
```

- You can now check the status of all Hadoop services using the jps command:


```
jps
```

```
hadoop@sanjay-VirtualBox:~/hadoop$ jps
7235 NodeManager
6677 DataNode
7593 Jps
6554 NameNode
7116 ResourceManager
6893 SecondaryNameNode
hadoop@sanjay-VirtualBox:~/hadoop$
```

Step 13 : Access Hadoop Namenode and Resource Manager :

- First we need to know our ip address, In Ubuntu we need to install net-tools to run ipconfig command, If you installing net-tools for the first time switch to default user :

```
sudo apt install net-tools
```

- Then run **ifconfig** command to know our ip address:

```
ifconfig
```

```
hadoop@sanjay-VirtualBox:~/hadoop$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.6 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 2401:4900:1c28:46c4:f76c:b206:abe3:2d45 prefixlen 64 scopeid 0x0<global>
    inet6 2401:4900:1c28:46c4:ed13:53f4:5c05:50c6 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::112b:300a:9242:51f3 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:83:31:35 txqueuelen 1000 (Ethernet)
    RX packets 645228 bytes 934388358 (934.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 93618 bytes 8998032 (8.9 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

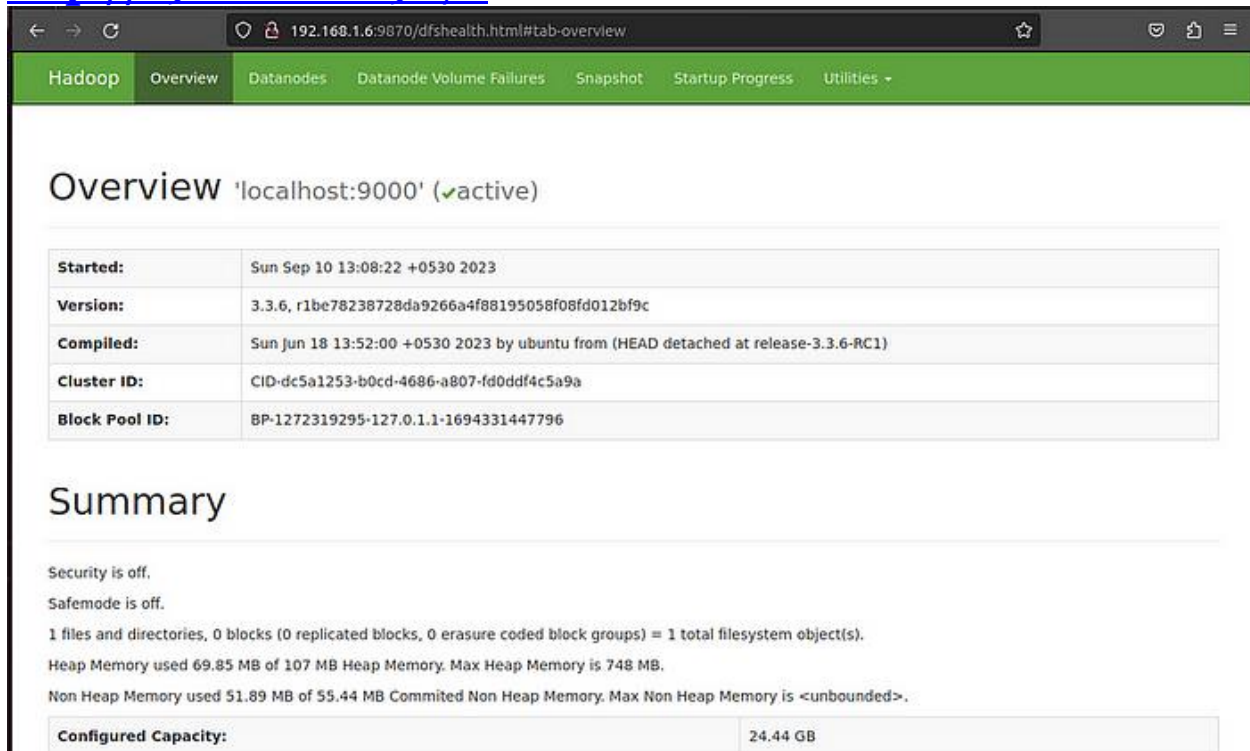
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 3331 bytes 491873 (491.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3331 bytes 491873 (491.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

hadoop@sanjay-VirtualBox:~/hadoop$
```


Here my ip address is 192.168.1.6.

- To access the **Namenode**, open your web browser and visit the URL <http://your-server-ip:9870>. You should see the following screen:

<http://192.168.1.6:9870>



The screenshot shows the Hadoop Namenode Overview page in a web browser. The browser's address bar displays the URL 192.168.1.6:9870/dfshealth.html#tab-overview. The page has a green navigation bar with tabs: Hadoop, Overview (selected), Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, and Utilities. The main content area is titled 'Overview 'localhost:9000' (✓active)'. Below the title is a table with the following information:

Started:	Sun Sep 10 13:08:22 +0530 2023
Version:	3.3.6, r1be78238728da9266a4f88195058f08fd012bf9c
Compiled:	Sun Jun 18 13:52:00 +0530 2023 by ubuntu from (HEAD detached at release-3.3.6-RC1)
Cluster ID:	CID-dc5a1253-b0cd-4686-a807-fd0ddf4c5a9a
Block Pool ID:	BP-1272319295-127.0.1.1-1694331447796

Below the table is a 'Summary' section. It contains the following text:

Security is off.
Safemode is off.
1 files and directories, 0 blocks (0 replicated blocks, 0 erasure coded block groups) = 1 total filesystem object(s).
Heap Memory used 69.85 MB of 107 MB Heap Memory. Max Heap Memory is 748 MB.
Non Heap Memory used 51.89 MB of 55.44 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

At the bottom, there is a table with the following information:

Configured Capacity:	24.44 GB
----------------------	----------

- To access Resource Manage, open your web browser and visit the URL <http://your-server-ip:8088>. You should see the following screen:

<http://192.168.1.6:8088>

Step 13 :Verify the Hadoop Cluster :

At this point, the Hadoop cluster is installed and configured. Next, we will create some directories in the HDFS filesystem to test the Hadoop.

- Let's create some directories in the HDFS filesystem using the following command:

```
hdfs dfs -mkdir /test1
hdfs dfs -mkdir /logs
```

- Next, run the following command to list the above directory:

```
hdfs dfs -ls /
```

You should get the following output:

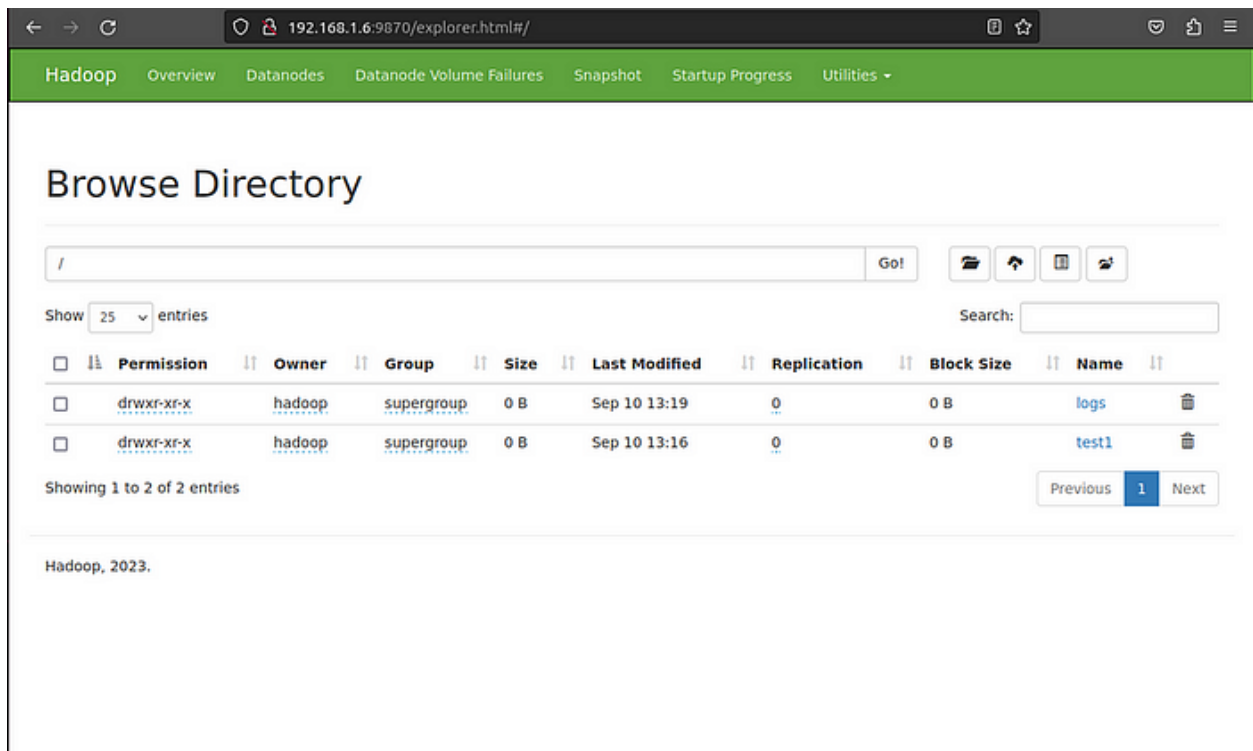
```
hadoop@sanjay-VirtualBox:~/hadoop$ hdfs dfs -ls /
Found 2 items
drwxr-xr-x - hadoop supergroup      0 2023-09-10 13:16 /logs
drwxr-xr-x - hadoop supergroup      0 2023-09-10 13:16 /test1
hadoop@sanjay-VirtualBox:~/hadoop$
```

- Also, put some files to **hadoop** file system. For the example, putting log files from host machine to **hadoop** file system.

```
hdfs dfs -put /var/log/* /logs/
```

You can also verify the above files and directory in the Hadoop web interface.

Go to the web interface, click on the Utilities => Browse the file system. You should see your directories which you have created earlier in the following screen:



Step 14 : To stop hadoop services :

To stop the Hadoop service, run the following command as a hadoop user:

```
stop-all.sh
```

```
hadoop@sanjay-VirtualBox:~/hadoop$ stop-all.sh
WARNING: Stopping all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [sanjay-VirtualBox]
Stopping nodemanagers
Stopping resourcemanager
hadoop@sanjay-VirtualBox:~/hadoop$
```