

Exp No: 9

Date:

HADOOP

SET UP A SINGLE HADOOP CLUSTER AND SHOW THE PROCESS USING WEB UI

AIM:

To set-up one node Hadoop cluster.

PROCEDURE:

1. System Update
2. Install Java
3. Add a dedicated Hadoop user
4. Install SSH and setup SSH certificates
5. Check if SSH works
6. Install Hadoop
7. Modify Hadoop config files
8. Format Hadoop filesystem
9. Start Hadoop 10. Check Hadoop through web UI 11. Stop Hadoop

THEORY

Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. A Hadoop frame-worked application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from a single server to thousands of machines, each offering local computation and storage.

HADOOP ARCHITECTURE

Hadoop framework includes following four modules.

NAME:HARISH RAJAN

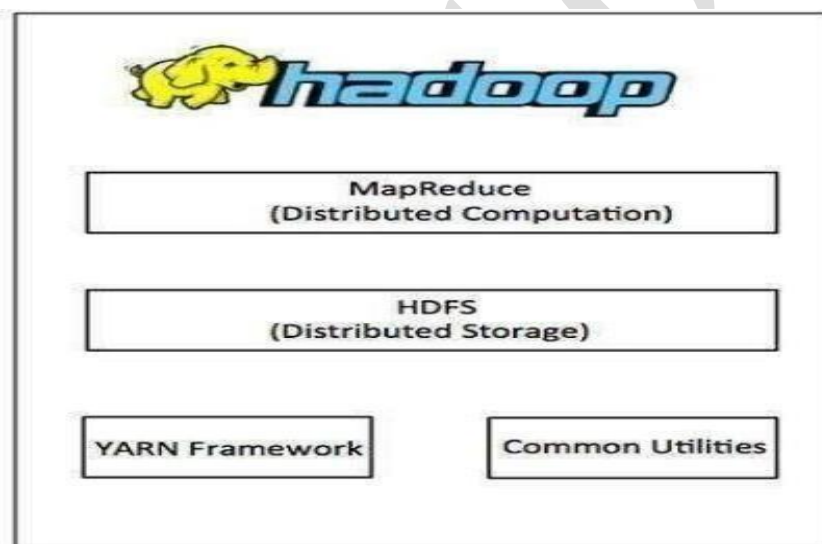
Hadoop Common: These are Java libraries and utilities required by other Hadoop modules. These libraries provide filesystem and OS level abstractions and contain the necessary Java files and scripts required to start Hadoop.

Hadoop YARN: This is a framework for job scheduling and cluster resource management.

Hadoop Distributed File System (HDFS): A distributed file system that provides highthroughput access to application data.

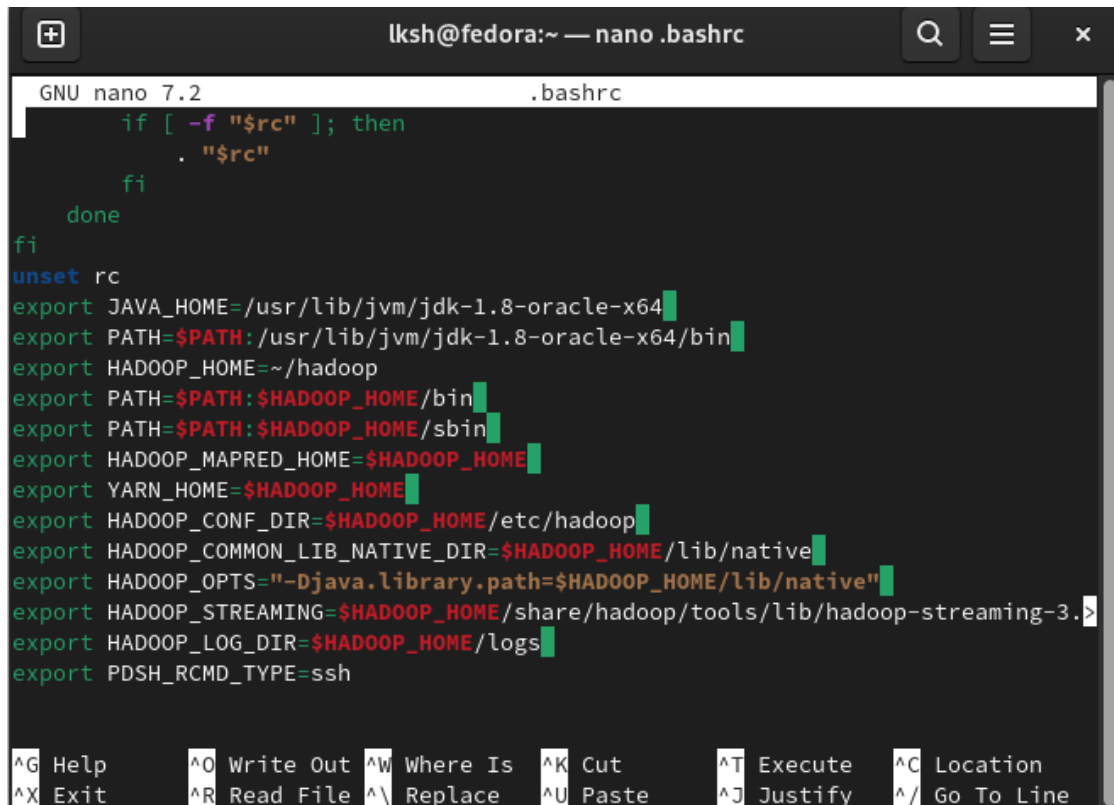
Hadoop MapReduce: This is a YARN-based system for parallel processing of large data sets.

We can use following diagram to depict these four components available in Hadoop framework.



PROCEDURE:

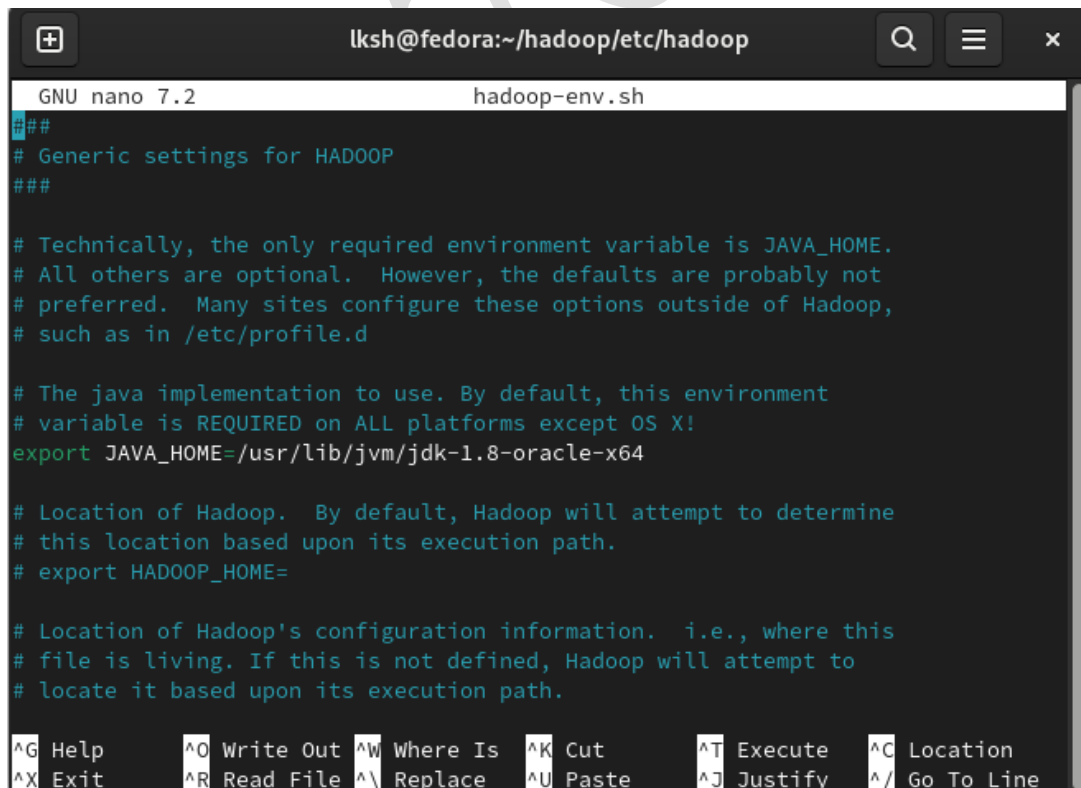
\$ nano ~/.bashrc

A screenshot of a terminal window showing the nano text editor editing the file ~/.bashrc. The window title is 'lksh@fedora:~ — nano .bashrc'. The editor shows a script with conditional sourcing of rc files and a series of export statements for Hadoop environment variables. The bottom status bar displays nano shortcuts.

```
GNU nano 7.2 .bashrc
if [ -f "$rc" ]; then
    . "$rc"
fi
done
fi
unset rc
export JAVA_HOME=/usr/lib/jvm/jdk-1.8-oracle-x64
export PATH=$PATH:/usr/lib/jvm/jdk-1.8-oracle-x64/bin
export HADOOP_HOME=~/.hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export HADOOP_STREAMING=$HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.0.0
export HADOOP_LOG_DIR=$HADOOP_HOME/logs
export PDSH_RCMD_TYPE=ssh

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

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

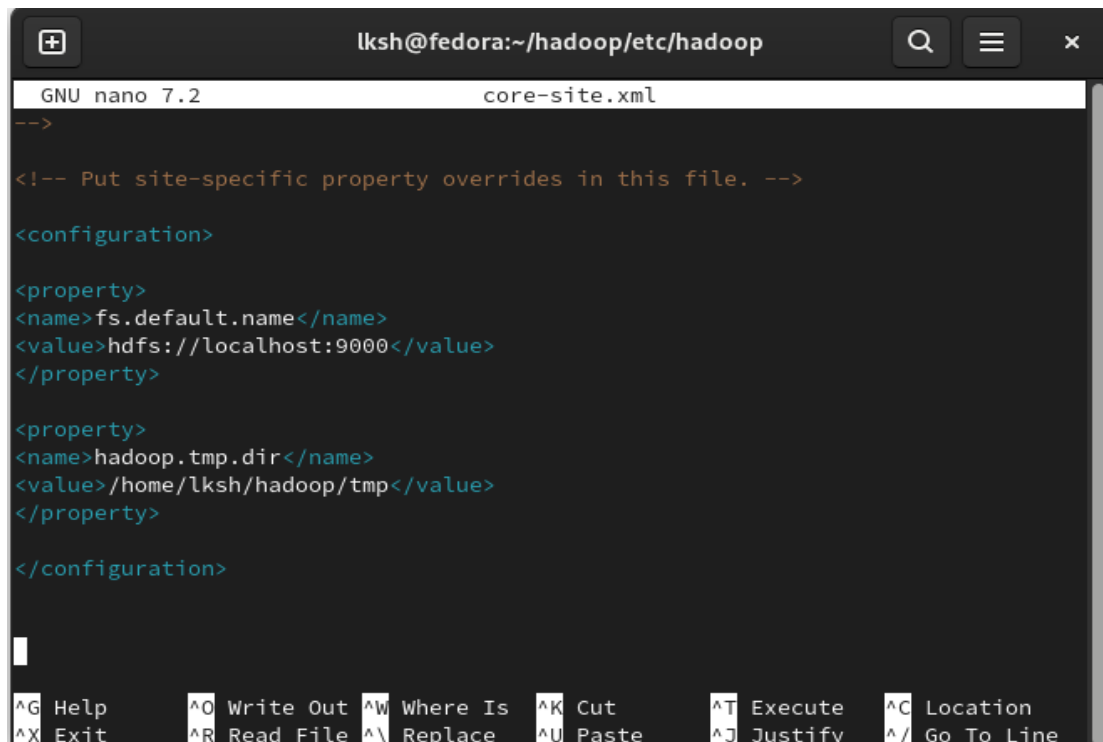
A screenshot of a terminal window showing the nano text editor editing the file \$HADOOP_HOME/etc/hadoop/hadoop-env.sh. The window title is 'lksh@fedora:~/hadoop/etc/hadoop'. The editor shows generic settings for Hadoop, including comments and an export statement for JAVA_HOME. The bottom status bar displays nano shortcuts.

```
GNU nano 7.2 hadoop-env.sh
##
# 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!
export JAVA_HOME=/usr/lib/jvm/jdk-1.8-oracle-x64
#
# Location of Hadoop.  By default, Hadoop will attempt to determine
# this location based upon its execution path.
# export HADOOP_HOME=
#
# Location of Hadoop's configuration information.  i.e., where this
# file is living.  If this is not defined, Hadoop will attempt to
# locate it based upon its execution path.

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

NAME:HARISH RAJAN

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

A screenshot of a terminal window showing the nano text editor editing the file core-site.xml. The window title is 'lksh@fedora:~/hadoop/etc/hadoop'. The editor shows the following XML content:

```
-->

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

<configuration>

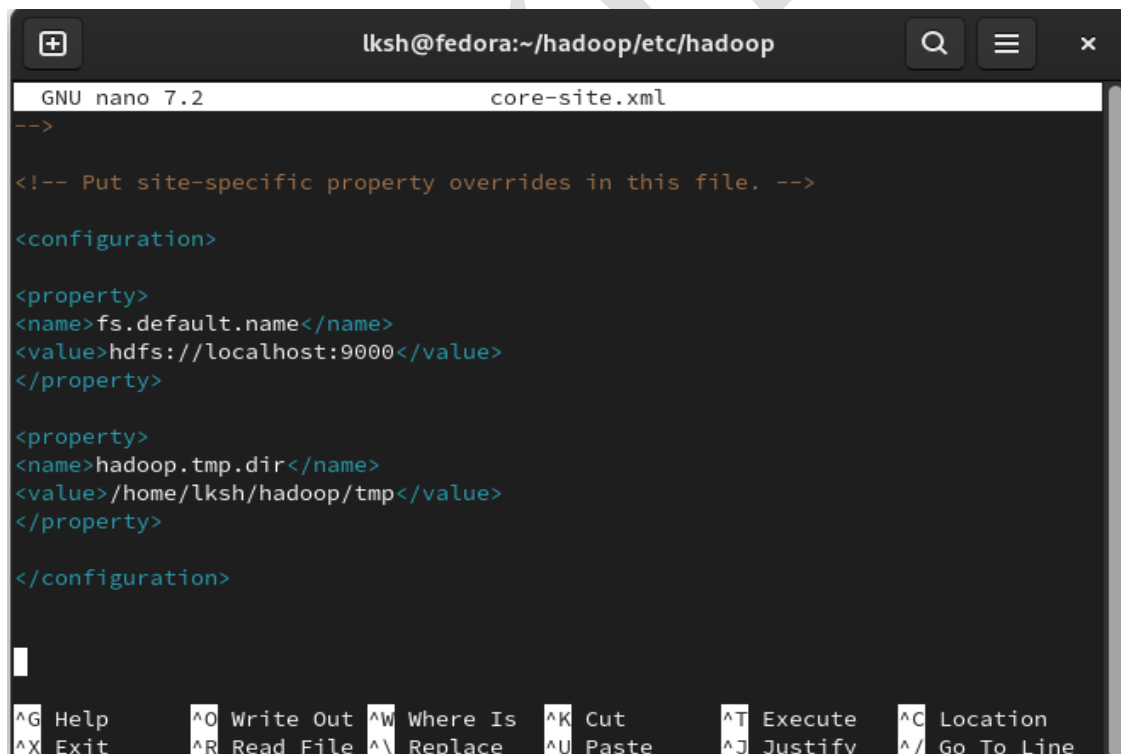
<property>
<name>fs.default.name</name>
<value>hdfs://localhost:9000</value>
</property>

<property>
<name>hadoop.tmp.dir</name>
<value>/home/lksh/hadoop/tmp</value>
</property>

</configuration>
```

The bottom status bar of the nano editor displays various keyboard shortcuts: ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Location, ^X Exit, ^R Read File, ^_ Replace, ^U Paste, ^J Justify, and ^_ Go To Line.

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

A screenshot of a terminal window showing the nano text editor editing the file hdfs-site.xml. The window title is 'lksh@fedora:~/hadoop/etc/hadoop'. The editor shows the following XML content:

```
-->

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

<configuration>

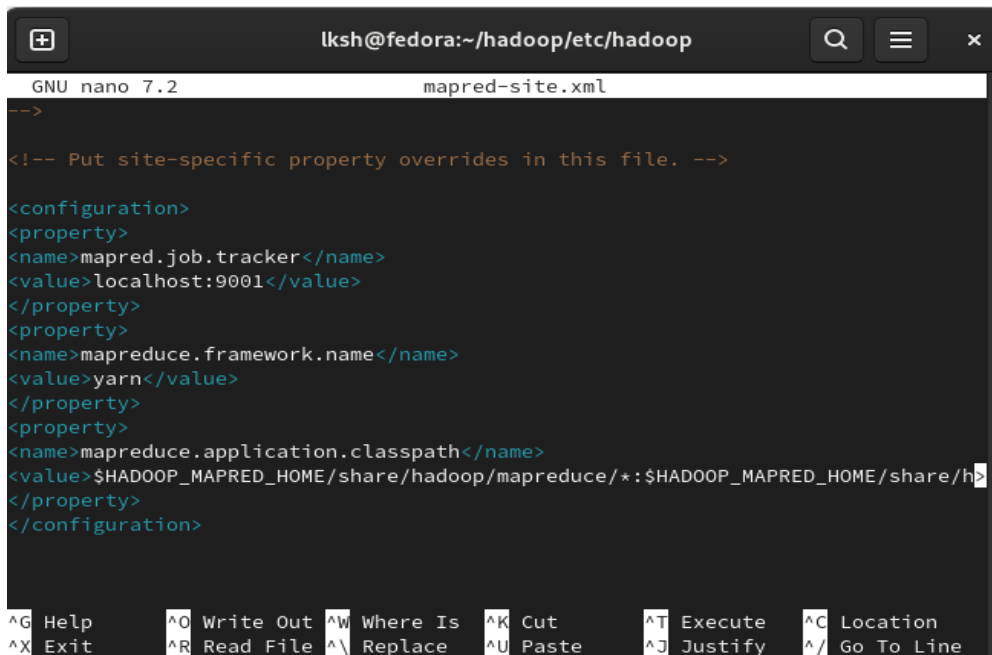
<property>
<name>fs.default.name</name>
<value>hdfs://localhost:9000</value>
</property>

<property>
<name>hadoop.tmp.dir</name>
<value>/home/lksh/hadoop/tmp</value>
</property>

</configuration>
```

The bottom status bar of the nano editor displays various keyboard shortcuts: ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Location, ^X Exit, ^R Read File, ^_ Replace, ^U Paste, ^J Justify, and ^_ Go To Line.

\$nano \$SHADOOP_HOME/etc/hadoop/mapred-site.xml

A screenshot of a terminal window showing the nano text editor editing the file mapred-site.xml. The window title is 'lksh@fedora:~/hadoop/etc/hadoop'. The editor shows XML configuration for MapReduce, including properties for the job tracker, framework name (yarn), and application classpath. The bottom status bar displays various nano editor shortcuts.

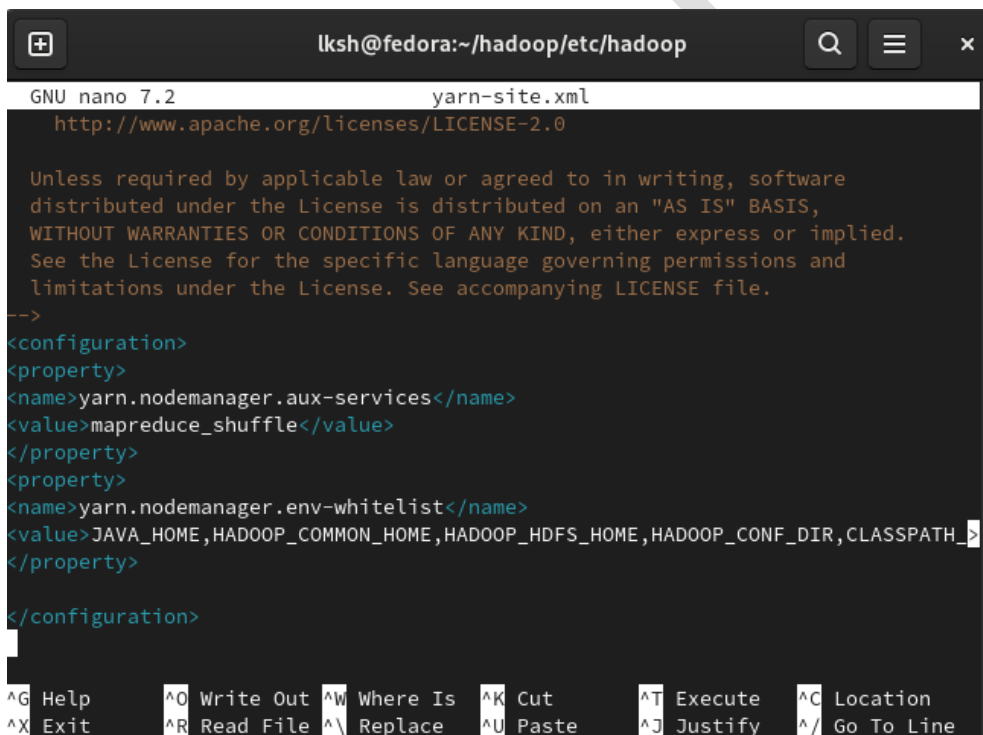
```
GNU nano 7.2 mapred-site.xml
-->

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

<configuration>
<property>
<name>mapred.job.tracker</name>
<value>localhost:9001</value>
</property>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
<property>
<name>mapreduce.application.classpath</name>
<value>$SHADOOP_MAPRED_HOME/share/hadoop/mapreduce/*:$SHADOOP_MAPRED_HOME/share/h
</property>
</configuration>

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

\$nano \$SHADOOP_HOME/etc/hadoop/yarn-site.xml

A screenshot of a terminal window showing the nano text editor editing the file yarn-site.xml. The window title is 'lksh@fedora:~/hadoop/etc/hadoop'. The editor shows the Apache License header and XML configuration for Yarn, including properties for auxiliary services and environment whitelist. The bottom status bar displays various nano editor shortcuts.

```
GNU nano 7.2 yarn-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.
-->

<configuration>
<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_
</property>
</configuration>

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

\$ start-all.sh

```
lksh@fedora:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as lksh 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 [fedora]
Starting resourcemanager
Starting nodemanagers
```

\$ jps

```
lksh@fedora:~/hadoop/etc/hadoop$ jps
3490 DataNode
4163 NodeManager
4599 Jps
3321 NameNode
4027 ResourceManager
3695 SecondaryNameNode
```

localhost:9870

The screenshot shows a web browser window with the address bar displaying 'localhost:9870/dfshealth.html#tab-overview'. The page has a green navigation bar with tabs: 'Hadoop', 'Overview', 'Datanodes', 'Datanode Volume Failures', 'Snapshot', 'Startup Progress', and 'Utilities'. The 'Overview' tab is selected, showing the 'Overview' page for 'localhost:9000' (active). Below the title is a table with the following information:

Started:	Thu Aug 15 16:47:15 +0530 2024
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-4797107d-bd90-4038-afed-e5c495ecd59b
Block Pool ID:	BP-1847114295-127.0.1.1-1723720483718

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 79.95 MB of 261 MB Heap Memory. Max Heap Memory is 871.5 MB.
Non Heap Memory used 47.19 MB of 48.65 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

At the bottom, there is a table with two columns: 'Configured Capacity:' and '0 B'.

localhost:8088

All Applications

localhost:8088/cluster



Cluster

About

Nodes

Node Labels

Applications

NEW

NEW SAVING

SUBMITTED

ACCEPTED

RUNNING

FINISHED

FAILED

KILLED

Scheduler

Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running
0	0	0	0	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes
1	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>

Show 20 entries

ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime
Showing 0 to 0 of 0 entries									

RESULT:

Thus, Hadoop has been successfully installed.

NAME:HARISH RAJAN