

EXP NO: 1(B)	Understanding different Hadoop modes Startup scripts, Configuration files.
DATE:	

AIM: -

BACKGROUND THEORY: -

PROCEDURE: -

- Switch to superuser mode using `sudo su`.
- Update all packages with `sudo yum update -y`.
- Install Java with `sudo yum install java-11-openjdk-devel -y` or `sudo yum install java-1.8.0-amazon-corretto-devel -y`.
- Verify the Java installation using `java -version`.
- Navigate to `/usr/local/` using `cd /usr/local/`.
- Download Hadoop with `sudo wget https://downloads.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz`.
- Extract the Hadoop archive using `sudo tar -xvzf hadoop-3.3.6.tar.gz`.
- Rename the extracted folder with `sudo mv hadoop-3.3.6 hadoop`.
- Edit the `.bashrc` file to add Hadoop and Java environment variables.
- Reload the `.bashrc` file with `source ~/.bashrc` and verify Hadoop installation using `hadoop version`.
- Edit `core-site.xml` to set the default file system and temporary directory.
- Configure `hdfs-site.xml` to set replication factor, NameNode, and DataNode directories.
- Create `mapred-site.xml` and configure the MapReduce framework to use YARN.
- Edit `yarn-site.xml` to set up ResourceManager and NodeManager services.
- Create the necessary Hadoop directories for the NameNode, DataNode, and temporary storage.
- Format the Hadoop HDFS filesystem using `hdfs namenode -format`.
- Start the HDFS services with `start-dfs.sh`.
- Start YARN services with `start-yarn.sh`.
- Check running processes with `jps` to ensure services are active.
- Access Hadoop web interfaces: NameNode at `http://localhost:9870/` and ResourceManager at `http://localhost:8088/`.

CODING: -

- `sudo su`
- `sudo yum update -y`
- `sudo yum install java-11-openjdk-devel -y` **or** `sudo yum install java-1.8.0-amazon-corretto-devel -y`
- `java -version`
- `cd /usr/local/`
- `sudo wget https://downloads.apache.org/hadoop/common/hadoop-3.3.6/hadoop-3.3.6.tar.gz`
- `sudo tar -xvzf hadoop-3.3.6.tar.gz`
- `sudo mv hadoop-3.3.6 hadoop`

- `sudo nano ~/.bashrc`
 - `# Hadoop variables`
 - `export HADOOP_HOME=/usr/local/hadoop`
 - `export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin`
 - `# Java variables`
 - `export JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:/bin/java::")`
 - `export PATH=$PATH:$JAVA_HOME/bin`
- `source ~/.bashrc`
- `hadoop version`
- `Configure core-site.xml`
 - `sudo nano $HADOOP_HOME/etc/hadoop/core-site.xml`
 - `<configuration>`
 - `<property>`
 - `<name>fs.defaultFS</name>`
 - `<value>hdfs://localhost:9000</value>`
 - `</property>`
 - `<property>`
 - `<name>hadoop.tmp.dir</name>`
 - `<value>/usr/local/hadoop/tmp</value>`
 - `<description>Temporary directory for Hadoop</description>`
 - `</property>`
 - `</configuration>`
- `Configure hdfs-site.xml`
 - `sudo nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml`
 - `<configuration>`
 - `<property>`
 - `<name>dfs.replication</name>`
 - `<value>1</value> <!-- Since this is a single-node setup -->`
 - `</property>`

- `<property>`
- `<name>dfs.namenode.name.dir</name>`
- `<value>file:///usr/local/hadoop/hdfs/namenode</value>`
- `</property>`
- `<property>`
- `<name>dfs.datanode.data.dir</name>`
- `<value>file:///usr/local/hadoop/hdfs/datanode</value>`
- `</property>`
- `</configuration>`
- Create the mapred-site.xml[If the mapred-site.xml.template is not present]
 - `sudo nano /usr/local/hadoop/etc/hadoop/mapred-site.xml`
 - `<configuration>`
 - `<property>`
 - `<name>mapreduce.framework.name</name>`
 - `<value>yarn</value>`
 - `</property>`
 - `</configuration>`
- Configure yarn-site.xml
 - `sudo nano $HADOOP_HOME/etc/hadoop/yarn-site.xml`
 - `<configuration>`
 - `<property>`
 - `<name>yarn.nodemanager.aux-services</name>`
 - `<value>mapreduce_shuffle</value>`
 - `</property>`
 - `<property>`
 - `<name>yarn.resourcemanager.resource-tracker.address</name>`
 - `<value>localhost:8025</value>`
 - `</property>`
 - `<property>`

- `<name>yarn.resourcemanager.scheduler.address</name>`
- `<value>localhost:8030</value>`
- `</property>`
- `<property>`
- `<name>yarn.resourcemanager.address</name>`
- `<value>localhost:8050</value>`
- `</property>`
- `</configuration>`
- Set Up Hadoop Directories
 - `sudo mkdir -p /usr/local/hadoop/hdfs/namenode`
 - `sudo mkdir -p /usr/local/hadoop/hdfs/datanode`
 - `sudo mkdir -p /usr/local/hadoop/tmp`
- Format the HDFS Filesystem
 - `hdfs namenode -format`
- Start Hadoop Services(As Non - Root User)
 - `start-dfs.sh`
 - `start-yarn.sh`
 - `Jps`
- Access the Hadoop Web Interfaces
 - **NameNode:** `http://localhost:9870/` (shows the HDFS overview)
 - **ResourceManager:** `http://localhost:8088/` (shows the YARN overview)
 - **LocalHost - 127.0.0.1 or public DNS :- 3.117.182.16**

OUTPUT: -

```
aws Services Search [Alt+S]
GNU nano 5.8 /root/.bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific environment
if [ [ "$PATH" != "$HOME/.local/bin:$HOME/bin:" ] ]
then
    PATH="$HOME/.local/bin:$HOME/bin:$PATH"
fi
export PATH

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions

alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'

[ Read 31 lines ]
Help Write Out Where Is Cut Execute Location Undo Set Mark To Bracket Previous
Exit Read File Replace Paste Justify Go To Line Redo Copy Where Was Next
```

```
aws Services Search [Alt+S]

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

For security reasons, the password you type will not be visible.

[sudo] password for hadoop:
hadoop is not in the sudoers file.
[hadoop@ip-172-31-40-166 ~]$ start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ip-172-31-40-166.ec2.internal]
[hadoop@ip-172-31-40-166 ~]$ start-yarn.sh
Starting resourcemanager
resourcemanager is running as process 33674. Stop it first and ensure /tmp/hadoop-hadoop-resourcemanager.pid file is empty before retry.
Starting nodemanagers
[hadoop@ip-172-31-40-166 ~]$ jps
35073 Jps
34913 NodeManager
34360 DataNode
34617 SecondaryNameNode
34250 NameNode
33674 ResourceManager
[hadoop@ip-172-31-40-166 ~]$

i-015e997a4b9339fce (BDA-Lab)
PublicIPs: 3.88.65.175 PrivateIPs: 172.31.40.166
```

```
# 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!
# export JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:/bin/java:/" )

# Location of Hadoop. By default, Hadoop will attempt to determine
# this location based upon its execution path.
```