EXPERIMENT-1

Aim:
To study the architecture of the Apache Hadoop framework and perform the installation and configuration of a Single-Node Hadoop Cluster using WSL (Windows Subsystem for Linux).

Theory:
Apache Hadoop is an open-source software framework used for distributed storage and processing of datasets of big data using the MapReduce programming model. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Since Hadoop is native to Linux, **WSL** allows Windows users to run a GNU/Linux environment directly on Windows, unmodified, without the overhead of a traditional virtual machine or dual-boot setup.

Key Components:

1. **HDFS (Hadoop Distributed File System):** A distributed file system that provides high-throughput access to application data.
2. **MapReduce:** A software framework for distributed processing of large data sets on compute clusters.
3. **YARN (Yet Another Resource Negotiator):** A framework for job scheduling and cluster resource management.
4. **Hadoop Common:** The common utilities that support the other Hadoop modules.

Architecture:
Hadoop follows a **Master-Slave Architecture**:

- **HDFS Architecture:**
  - NameNode (Master): Manages the file system namespace and controls access to files.
  - DataNodes (Slaves): Manage storage attached to the nodes that they run on.
- **YARN Architecture:**
  - ResourceManager (Master): Managing resources across the cluster.
  - NodeManager (Slaves): Launches and monitors containers (tasks).

Description Include install:
This experiment involves setting up a "Pseudo-Distributed Mode" (Single Node Cluster) where all Hadoop daemons (NameNode, DataNode, ResourceManager, etc.) run on a single machine (the WSL instance) as separate Java processes.

Pre-requisites:

1. Windows 10 or 11 with WSL 2 enabled.

2. Ubuntu (or another distro) installed from the Microsoft Store.
3. Java Development Kit (JDK 8 is recommended).
4. SSH (Secure Shell) for managing Hadoop daemons.

Step-by-step installation:

1. Update System and Install Java:
Open the Ubuntu terminal in WSL and run:

```
sudo apt update && sudo apt upgrade
sudo apt install openjdk-8-jdk
java -version
```

2. Configure SSH (Passphraseless Login):
Hadoop requires SSH to manage its nodes.

```
sudo apt install openssh-server
sudo service ssh start
ssh-keygen -t rsa -P ""
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
ssh localhost
exit
```

3. Download and Extract Hadoop:

```
wget https://downloads.apache.org/hadoop/common/hadoop-3.3.6/hadoop-
3.3.6.tar.gz
tar -xzvf hadoop-3.3.6.tar.gz
mv hadoop-3.3.6 hadoop
```

4. Configure Environment Variables:
Edit the bash configuration file: `nano ~/.bashrc`
Add the following lines at the end:

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export HADOOP_HOME=~/hadoop
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
```

Save and exit (Ctrl+O, Enter, Ctrl+X), then run: `source ~/.bashrc`

5. Edit Hadoop Configuration Files:
Navigate to the config directory: `cd $HADOOP_HOME/etc/hadoop`

- **hadoop-env.sh:** Find the `JAVA_HOME` line and change it to:
  `export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64`
- **core-site.xml:** Add between <configuration> tags:

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

- **hdfs-site.xml:** Add between <configuration> tags:

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

6. Format the NameNode:

Initialize the file system (do this only once):

```
hdfs namenode -format
```

7. Start Hadoop Services:

```
start-dfs.sh
start-yarn.sh
```

8. Verify Installation:
Run the `jps` command. You should see the following processes:

- NameNode
- DataNode
- ResourceManager
- NodeManager
- SecondaryNameNode
- Jps

Conclusion:

In this experiment, we successfully studied the architecture of Apache Hadoop and installed it on the Windows Subsystem for Linux (WSL). We configured the environment variables and XML files required for a single-node cluster. The successful startup of all Hadoop daemons (verified via the `jps` command) confirms that the environment is ready for executing MapReduce jobs.