BIG DATA MANAGEMENT:
Hadoop Single Node Cluster Setup

CZ4123

Contents

0. Prerequisite

1. Linux Operating System (Ubuntu 22.04)
   - How to run Ubuntu Desktop on a virtual machine using VirtualBox
   - Ubuntu image
   - Linux command line for beginners
2. Install Java 8

   ```
   sudo apt install openjdk-8-jdk-headless
   ```

3. Download Hadoop 3.3.1 binary file

   ```
   wget
   https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
   ```

Hadoop Single Node Cluster Setup

Introduction on Hadoop

Hadoop

- Open-source framework that allows for the distributed processing of large data sets across clusters of computers
- **Scalability**: easy to add new hardware to cluster
- **Reliability**: single machines can fail and be handled
- Application: Facebook Messenger, Walmart's inventory recommendation, LinkedIn's big data analytics …

Apache Hadoop, Hadoop Use Cases | Project Pro

1.1. Install Packages

1. Install Java 8

```
sudo apt install openjdk-8-jdk-headless
```

2. Install SSH tools

```
sudo apt install ssh pdsh
```

3. Install Hadoop

```
cd ~/Downloads
wget https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
tar xzf hadoop-3.3.1.tar.gz
sudo mv hadoop-3.3.1/usr/share/hadoop
```

Hadoop modes

- **Local (Standalone) Mode**: on a single node, non-distributed mode, as a single Java process
- **Pseudo-distributed Mode**: on a single node, each Hadoop daemon runs in a separate Java process
- **Fully-Distributed Mode**: a few nodes to extremely large clusters with thousands of nodes, connected by network

Hadoop: Setting up a Single Node Cluster, Hadoop Cluster Setup

1.2. Configure System Environment

1. Edit system path in ~/.bashrc:

```
gedit ~/.bashrc
export PDSH_RCMD_TYPE=ssh
export HADOOP_HOME=/usr/share/hadoop
export PATH=$PATH:$HADOOP_HOME:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
```

2. Reopen terminal

3. Edit Java path in HADOOP_HOME/etc/hadoop/hadoop-env.sh L54:

```
gedit $HADOOP_HOME/etc/hadoop/hadoop-env.sh
export JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64/"
```

4. Run Hadoop (non-distributed mode)

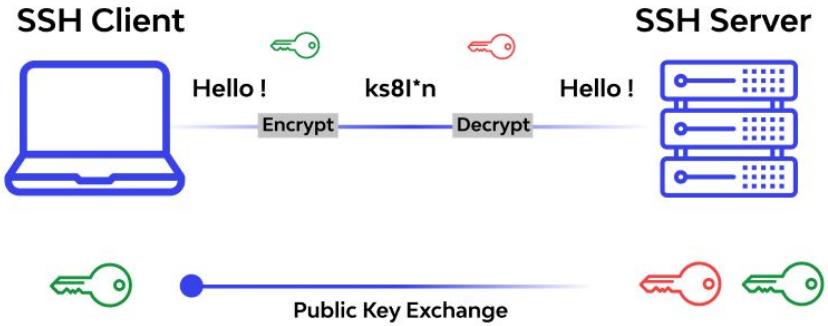
```
hadoop jar /usr/share/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.1.jar pi
16 1000
```

Hadoop Single Node Cluster Setup

Linux Network Setup

SSH Basics

- **SSH** (Secure Shell Protocol): secure network services over an unsecured network
- **Asymmetric cryptography**:
  - **Public Key** can be shared and used to encrypt message
  - Message can only be decrypted by **Private Key**



What Is SSH Protocol?

2. Configure System Network

1. Initialize SSH settings in ~/.ssh

```
cd ~/.ssh
rm ./id_rsa*
```

2. Generate a new SSH public key

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

3. Authorize public key

```
ssh-copy-id -i ~/.ssh/id_rsa.pub localhost
```

4. Test connection

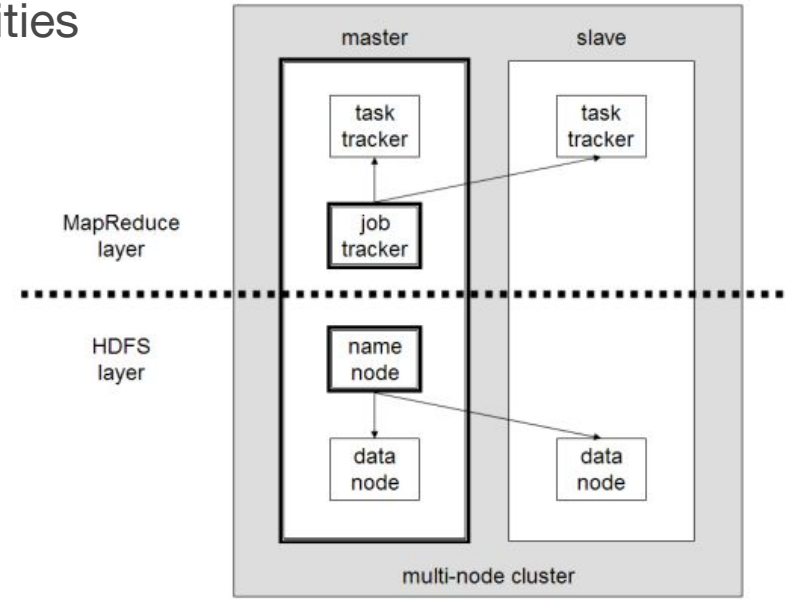
```
ssh localhost
exit
```

Hadoop Single Node Cluster Setup

Hadoop Configuration

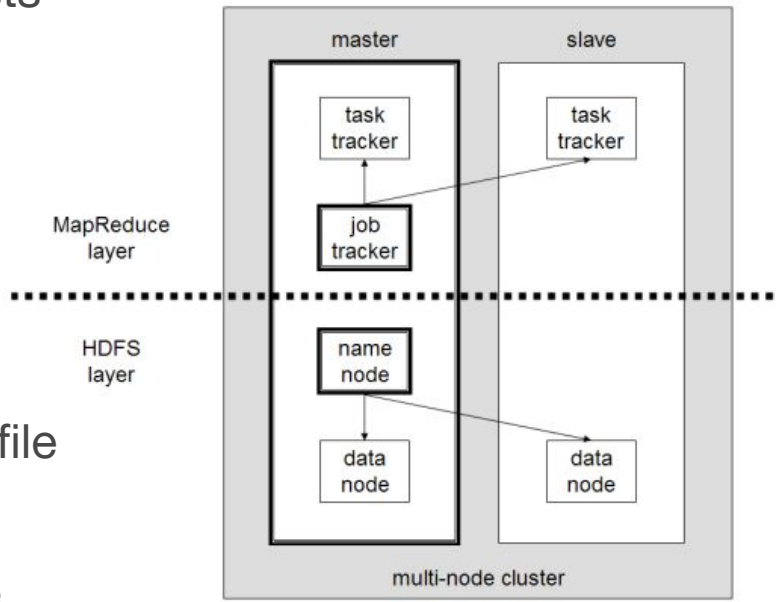
Hadoop modules

- **Hadoop Common**: core libraries and utilities
- **HDFS** (Hadoop Distributed File System): distributed file-system that stores data
- **YARN** (Yet Another Resource Negotiator): computing resources manager and job scheduler
- **MapReduce**: YARN-based parallel data processor of large data sets



MapReduce

- **JobTracker**: receives MapReduce requests and talks to NameNode
- **TaskTracker (Mapper)**: performs actual processing code on data files



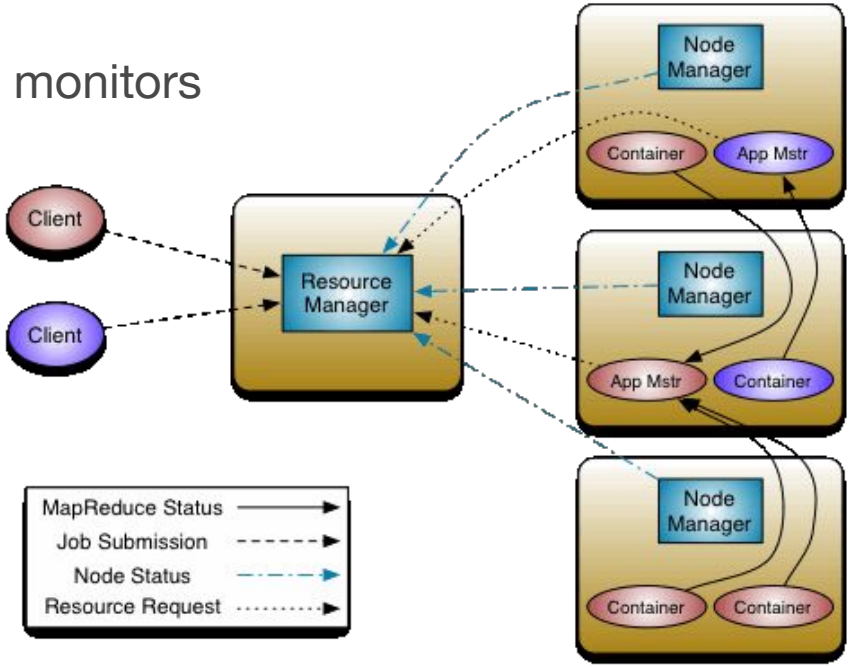
HDFS

- **NameNode**: a single node that manages file system and stores metadata in RAM
- **DataNodes**: multiple nodes that store the actual data into HDFS

[HDFS Architecture]

YARN

- **ResourceManager**: perform job tracking and resource allocation
- **NodeManager**: per-machine agent monitors progress of the execution



Apache Hadoop YARN

Hadoop configuration

Under HADOOP_HOME/etc/hadoop/:

- hadoop-env.sh: environment variables of JDK
- core-site.xml: runtime Hadoop environment settings
- hdfs-site.xml: HDFS settings (NameNode, DataNode)
- mapred-site.xml: MapReduce settings (framework, path)
- yarn-site.xml: YARN settings (NodeManager, ResourceManager)

3.1. Configure Hadoop

1. Add in HADOOP_HOME/etc/hadoop/core-site.xml:

```
gedit $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>file:/usr/share/hadoop/tmp</value>
    </property>
</configuration>
```

3.2. Configure HDFS

1. Add in HADOOP_HOME/etc/hadoop/hdfs-site.xml:

```
gedit $HADOOP_HOME/etc/hadoop/hdfs-site.xml
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
    <property>
        <name>dfs.namenode.name.dir</name>
        <value>file:/usr/share/hadoop/tmp/dfs/name</value>
    </property>
    <property>
        <name>dfs.datanode.data.dir</name>
        <value>file:/usr/share/hadoop/tmp/dfs/data</value>
    </property>
    <property>
        <name>dfs.namenode.datanode.registration.ip-hostname-check</name>
        <value>false</value>
    </property>
</configuration>
```

3.3. Configure MapReduce

1. Add in HADOOP_HOME/etc/hadoop/mapred-site.xml:

```
gedit $HADOOP_HOME/etc/hadoop/mapred-site.xml
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
    <property>
        <name>mapreduce.application.classpath</name>
        <value>$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/*:$HADOOP_
MAPRED_HOME/share/hadoop/mapreduce/lib/*</value>
    </property>
</configuration>
```

3.4. Configure YARN

1. Add in HADOOP_HOME/etc/hadoop/yarn-site.xml:

```
gedit $HADOOP_HOME/etc/hadoop/yarn-site.xml
<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_PREPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_HOME,PATH,LANG,TZ,HADOOP_MAPRED_HOME</value>
    </property>
</configuration>
```

Hadoop Single Node Cluster Setup

Hadoop Execution and Management

Hadoop commands

- shellcommand [COMMAND] [OPTIONS]
  - shellcommand: hadoop, hdfs, yarn
- hadoop jar <file_path> [ARGS] …
  - Runs a jar file
- hdfs namenode [COMMAND]
  - Perform operation on the NameNode
- hdfs dfs [COMMAND [COMMAND_OPTIONS]]
  - Run a filesystem command: -mkdir, -ls, -put, -cat, …

Hadoop Commands Guide , HDFS Commands Guide

4.1. Launch Hadoop

1. Execute in HADOOP_HOME:

```
hdfs namenode –format -force
start-dfs.sh
start-yarn.sh
```

2. To stop Hadoop

```
stop-all.sh
```

3. Monitor Java process

```
jps
```

4. Web interface

- Hadoop: http://localhost:9870/
- Yarn: http://localhost:8088/

4.2. Execute MapReduce Job

1. Prepare files

```
hdfs dfs -mkdir -p /user/hadoop
hdfs dfs -mkdir -p /user/input
hdfs dfs -put /usr/share/hadoop/etc/hadoop/*.xml /user/input
```

2. Run Hadoop (pseudo-distributed mode)

```
hadoop jar
/usr/share/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.1.jar
grep /user/input output 'dfs[a-z.]+'
hdfs dfs -cat /user/output/*
```

THANK YOU