Hadoop Installation Steps

1. Create Virtual Machine on Cloud Provider (Configure Firewall to allow port 8000 - 50000).
2. SSH into each Virtual Machine.
3. In each Virtual Machine, create a user "hadoop" with sudo privileges and switch to that user using following command.

```
sudo adduser hadoop
sudo adduser hadoop sudo
su - hadoop
```

4. Install Java OpenJDK 8

```
sudo apt-get update && sudo apt-get -y dist-upgrade && sudo apt-get -y install openjdk-8-jdk-headless
```

5. Generate SSH Key for all virtual machines.

```
ssh-keygen
```

6. Copy SSH Key that store in `~/.ssh/id_rsa.pub` and put it into `~/.ssh/authorized_keys` file of each virtual machine.
7. On master node, config `~/.ssh/config` file.

```
Host 10.148.0.3
    HostName 10.148.0.3
    User hadoop
    IdentityFile ~/.ssh/id_rsa
Host 10.138.0.2
    HostName 10.138.0.2
    User hadoop
    IdentityFile ~/.ssh/id_rsa
Host 10.152.0.2
    HostName 10.152.0.2
    User hadoop
    IdentityFile ~/.ssh/id_rsa
Host 10.166.0.3
    HostName 10.166.0.3
    User hadoop
    IdentityFile ~/.ssh/id_rsa
Host 10.188.0.3
    HostName 10.188.0.3
    User hadoop
    IdentityFile ~/.ssh/id_rsa
```

8. Download hadoop 3.3.1, extract it and rename it to hadoop.

```
wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
tar xvzf hadoop-3.3.1.tar.gz && mv hadoop-3.3.1 hadoop && rm hadoop-3.3.1.tar.gz
```

9. Configure Hadoop Environment Variables which locate at `~/hadoop/etc/hadoop/hadoop-env.sh` on all nodes.

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export HDFS_NAMENODE_USER="hadoop"
export HDFS_DATANODE_USER="hadoop"
export HDFS_SECONDARYNAMENODE_USER="hadoop"
export YARN_RESOURCEMANAGER_USER="hadoop"
export YARN_NODEMANAGER_USER="hadoop"
export HADOOP_HOME=/home/hadoop/hadoop
```

10. Activate Hadoop Environment Variables using following command.

```
source ~/hadoop/etc/hadoop/hadoop-env.sh
```

11. Create directory for hdfs data on all nodes and change owner of file to be `hadoop` using following command.

```
sudo mkdir -p /usr/local/hadoop/hdfs/data
sudo chown -R hadoop:hadoop /usr/local/hadoop/hdfs/data
```

12. Configure `core-site.xml` which locate at `~/hadoop/etc/hadoop/core-site.xml` on all nodes. (10.148.0.3 is my Master Node's internal IP)

```
<configuration>
  <property>
    <name>fs.default.name</name>
    <value>hdfs://10.148.0.3:9000/</value>
  </property>
</configuration>
```

13. Configure `hdfs-site.xml` which locate at `~/hadoop/etc/hadoop/hdfs-site.xml` on master node. (10.138.0.2 is internal IP of another node that we selected as SecondaryNameNode)

```
<configuration>
    <property>
      <name>dfs.replication</name>
      <value>3</value>
    </property>
    <property>
      <name>dfs.namenode.name.dir</name>
      <value>file:///usr/local/hadoop/hdfs/data</value>
    </property>
    <property>
      <name>dfs.secondary.http.address</name>
      <value>10.138.0.2:50090</value>
    </property>
</configuration>
```

14. Configure `yarn-site.xml` which locate at `~/hadoop/etc/hadoop/yarn-site.xml` on all nodes.

Master Node

```
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
  </property>
</configuration>
```

Worker Node

```
<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
  </property>
  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>10.148.0.3</value>
  </property>
</configuration>
```

15. Configure `mapred-site.xml` which locate at `~/hadoop/etc/hadoop/mapred-site.xml` on master node. (10.148.0.3 is my Master Node's internal IP)

```
<configuration>
    <property>
      <name>mapreduce.jobtracker.address</name>
      <value>10.148.0.3:54311</value>
    </property>
    <property>
      <name>mapreduce.framework.name</name>
      <value>yarn</value>
    </property>
    <property>
      <name>yarn.nodemanager.vmem-check-enabled</name>
      <value>false</value>
    </property>
    <property>
      <name>yarn.app.mapreduce.am.env</name>
      <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
    </property>
    <property>
      <name>mapreduce.map.env</name>
      <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
    </property>
    <property>
      <name>mapreduce.reduce.env</name>
      <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
    </property>
</configuration>
```

16. Configure masters file which locate at `~/hadoop/etc/hadoop/masters` on master node.

```
10.148.0.3
```

17. Configure workers file which locate at `~/hadoop/etc/hadoop/workers` on master node.

```
10.148.0.3
10.138.0.2
10.152.0.2
10.166.0.3
10.188.0.3
```

18. Configure alias command in `~/.bashrc` file and auto activate hadoop environment configuration file.

```
alias hadoop="~/hadoop/hadoop-3.1.4/bin/hadoop"
source ~/hadoop/etc/hadoop/hadoop-env.sh
```

19. Activate our `~/.bashrc` file.

```
source ~/.bashrc
```

20. Format HDFS.

```
sudo ~/hadoop/bin/hdfs namenode -format
```

21. Start all Hadoop Service

```
sudo ~/hadoop/sbin/start-all.sh
```

22. Check Service Status using following command.

```
jps
```