1. Install Prerequisites on Master

- **Create Alias for IpAddress.**

edit **/etc/hosts** file and add all ip-addresses with alias

for example add as mentioned below (remember use private IP of EC2 instances)

172.31.15.76 master

172.31.14.69 slave01

172.31.11.65 slave02

- **Install java 7** (recommended ORACLE)

Add Repository for java so that ii will get installed using apt-get. Execute following commands in order

i. **sudo apt-get install python-software-properties**
ii. **sudo add-apt-repository ppa:webupd8team/java**
iii. **sudo apt-get update**
iv. **sudo apt-get install oracle-java8-installer**

2. Setup SSH from master to all slaves

Execute following command

sudo apt-get install openssh-server openssh-client

generate keypairs

ssh-keygen -t rsa -P ""

configure password-less SSH

copy content of **.ssh/id_rsa.pub** to **.ssh/authorized_keys** of master and both slaves for doing ssh.

3. Install Hadoop On Master

Download Hadoop on master machine

wget http://archive.cloudera.com/cdh5/cdh/5/hadoop-2.6.0-cdh5.10.0.tar.gz

Untar the downloaded file

tar xzf hadoop-2.6.0-cdh5.10.0.tar.gz

edit **~/.bashrc** file and add following entires at last

```
export HADOOP_PREFIX="/home/ubuntu/hadoop-2.6.0-cdh5.10.0"

export PATH=$PATH:$HADOOP_PREFIX/bin

export PATH=$PATH:$HADOOP_PREFIX/sbin
```

```
export HADOOP_MAPRED_HOME=${HADOOP_PREFIX}
export HADOOP_COMMON_HOME=${HADOOP_PREFIX}
export HADOOP_HDFS_HOME=${HADOOP_PREFIX}
export YARN_HOME=${HADOOP_PREFIX}
```

HADOOP CONFIGURATION SETTINGS

Please go inside <HADOOP_HOME>/etc/Hadoop directory to list all xml files.

set JAVA_HOME in hadoop_env.sh file , for example - /usr/lib/jvm/java-7-oracle/

copy following in core-site.xml

```
<configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://master:9000</value>
    </property>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>/home/ubuntu/hdata</value>
    </property>
</configuration>
```

copy following in hdfs-site.xml

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>2</value>
    </property>
</configuration>
```

copy following in mapred-site.xml

```
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
```

```
            <value>yarn</value>
        </property>
    </configuration>
```

copy following in yarn-site.xml

```
    <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.resource-tracker.addrees</name>
            <value>master:8025</value>
        </property>
        <property>
            <name>yarn.resourcemanager.scheduler.address</name>
            <value>master:8030</value>
        </property>
        <property>
            <name>yarn.resourcemanager.address</name>
            <value>master:8040</value>
        </property>
    </configuration>
```

Add slaves information in slaves file.

4. Install Hadoop On Each Slaves

Install prerequisites on slaves as we did on master by following only Step -1 in each slave machine.

Then follow the below mentioned steps

copy already configured setup of hadoop from master to both slaves

First create tar in master

tar czf hadoop.tar.gz hadoop-2.6.0-cdh5.10.0

copy the above created tar into both slaves machine using following commands

scp hadoop.tar.gz slave01:~

scp hadoop.tar.gz slave02:~

Execute following command on both slaves machine

tar xzf hadoop.tar.gz

5. Start Hadoop Cluster

Execute following command on master

Format namenode

$HADOOP_HOME/bin/hdfs namenode -format

Start dfs

$HADOOP_HOME/sbin/start-dfs.sh

Start yarn

$HADOOP_HOME/sbin/start-yarn.sh

Execute **jps** in all master and slaves to see below mentioned processes to run

jps on master should list following processes

ResourceManager

NameNode

SecondaryNameNode

jps on each slave should list following

NodeManager

DataNode

Access Namenode UI using following URL

http://<Public_IP_Master_Machine>:50070

Access Resource Manager UI using following URL

http://<Public_IP_Master_Machine>:8088