

## Instalación de pseudo-cluster con Cloudera

Añadimos el repositorio

```
# wget -c
https://archive.cloudera.com/cdh5/redhat/6/x86_64/cdh/cloudera-cdh5.repo?_ga=1.2136130
12.1732162428.1467798696 -O /etc/yum.repos.d/cloudera.repo
# rpm --import
https://archive.cloudera.com/cdh5/redhat/5/x86_64/cdh/RPM-GPG-KEY-cloudera
```

Instalamos el paquete pseudo-distribuido

```
# yum install hadoop-conf-pseudo
```

Cambiamos el valor por defecto

```
sed -i "s/localhost:8020/kerberos.example.com:8020/" /etc/hadoop/conf/core-site.xml
```

Inicializamos el file system

```
# sudo -u hdfs hdfs namenode -format
```

Arrancamos los servicios de HDFS

```
# for x in `cd /etc/init.d ; ls hadoop-hdfs-*` ; do sudo service $x start ; done
```

Creamos las estructuras básicas de directorios

```
# /usr/lib/hadoop/libexec/init-hdfs.sh
```

Listamos el contenido de nuestro HDFS

```
# sudo -u hdfs hadoop fs -ls -R /
```

Modificamos la configuración de YARN

```
# sed -i "s/localhost:8021/kerberos.example.com:8021/" /etc/hadoop/conf/mapred-site.xml
# sed -i "s/localhost:10020/kerberos.example.com:10020/" /etc/hadoop/conf/mapred-site.xml
# sed -i "s/localhost:19888/kerberos.example.com:19888/" /etc/hadoop/conf/mapred-site.xml
```

Arrancamos los servicios de YARN

```
# service hadoop-yarn-resourcemanager start
# service hadoop-yarn-nodemanager start
# service hadoop-mapreduce-historyserver start
```

Creamos usuarios de sistema para trabajar con ellos

```
# adduser foouser ; passwd foouser
# adduser baruser ; passwd baruser
# adduser bofh ; passwd bofh
```

Creamos grupos y asignamos a los usuarios

```
# groupadd datamaster
# groupadd datasec
# groupadd dataultrasec
# usermod -a -G datasec foouser
# usermod -a -G dataultrasec baruser
# usermod -a -G datamaster bofh
```

Creamos sus directorios en HDFS

```
# sudo -u hdfs hdfs dfs -mkdir -p /user/{foouser,baruser,bofh}
# sudo -u hdfs hdfs dfs -chown foouser:foouser /user/foouser
# sudo -u hdfs hdfs dfs -chown baruser:baruser /user/baruser
# sudo -u hdfs hdfs dfs -chown bofh:bofh /user/bofh
```

Entramos al sistema con el usuario foouser

```
# su - foouser
```

Verificamos que el cluster funciona correctamente

```
$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar pi 1 1
```

Si añadimos en el fichero de hosts de nuestra maquina la linea con la IP y el nombre del servidor podremos usar los interfaces web para comprobar el estado del cluster.

EN NUESTRO PORTÁTIL (Con linux!!)

```
echo "IP kerberos.example.com" >> /etc/hosts
```

Puertos a los que podemos conectar para verificar que todo está funcionando

<http://kerberos.example.com:8088>

<http://kerberos.example.com:19888>

<http://kerberos.example.com:50070>

Ponemos el arranque automático para todos los servicios del cluster

```
# chkconfig hadoop-yarn-resourcemanager on
# chkconfig hadoop-yarn-nodemanager on
# chkconfig hadoop-mapreduce-historyserver on
# chkconfig hadoop-hdfs-namenode on
# chkconfig hadoop-hdfs-datanode on
# chkconfig hadoop-hdfs-secondarynamenode on
```

Realizamos unas pequeñas pruebas para verificar que el cluster funciona:

```
# hadoop fs -put /etc/services
# hadoop fs -ls
# hadoop fs -cat services
# hadoop fs -rm services
# hadoop fs -ls
```