

Práctica 1: Instalación de clusters Hadoop

Edgar Seijo Otero

Índice

Parte I: Instalación manual de un cluster.....	3
Objetivos.....	3
1. Máquina virtual base.....	3
2. Descarga e instalación de Hadoop.....	6
3. Pasos como usuario hdmaster.....	8
4. Configuración de los demonios de Hadoop.....	9
5. Crear el cluster hadoop.....	12
Parte II: Uso básico del cluster.....	15
1. Inicio de Hadoop.....	15
2. Uso como usuario no privilegiado.....	19
Parte III: Modificación del cluster.....	22
1. Añadir y retirar DataNodes/NodeManagers.....	22
2. Rack awareness.....	27

Parte I: Instalación manual de un cluster

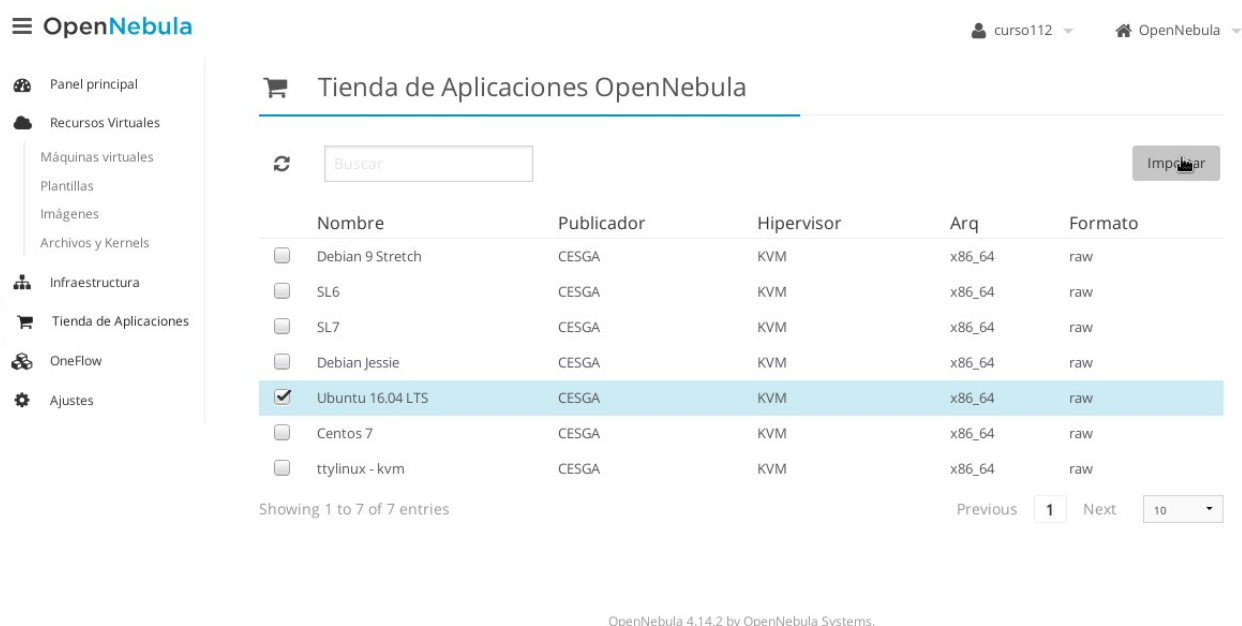
Objetivos

En la primera parte de este documento se instalará un cluster Hadoop utilizando 3 máquinas virtuales del cloud del CESGA. Para ello, se explicará el proceso de creación de las máquinas virtuales en el cloud y su posterior configuración como cluster Hadoop.

Al terminar esta parte, el cluster estará formado por un NameNode/ResourceManager, un DataNode/NodeManager/CheckpointNode y un DataNode/NodeManager/CheckpointNode.

1. Máquina virtual base

En primer lugar, se creará una máquina virtual base desde la que poder crear los nodos del cluster. Para ello, accedemos a <http://nebula.cesga.es/> y, desde la "Tienda de aplicaciones" importamos una "Ubuntu 16.04 LTS".



The screenshot shows the OpenNebula web interface. On the left is a sidebar with navigation links: Panel principal, Recursos Virtuales (Máquinas virtuales, Plantillas, Imágenes, Archivos y Kernels), Infraestructura, Tienda de Aplicaciones (highlighted), OneFlow, and Ajustes. The main area is titled 'Tienda de Aplicaciones OpenNebula' and contains a search bar, a refresh button, and an 'Importar' button. Below these is a table of available images:

	Nombre	Publicador	Hipervisor	Arq	Formato
<input type="checkbox"/>	Debian 9 Stretch	CESGA	KVM	x86_64	raw
<input type="checkbox"/>	SL6	CESGA	KVM	x86_64	raw
<input type="checkbox"/>	SL7	CESGA	KVM	x86_64	raw
<input type="checkbox"/>	Debian Jessie	CESGA	KVM	x86_64	raw
<input checked="" type="checkbox"/>	Ubuntu 16.04 LTS	CESGA	KVM	x86_64	raw
<input type="checkbox"/>	Centos 7	CESGA	KVM	x86_64	raw
<input type="checkbox"/>	ttylinux - kvm	CESGA	KVM	x86_64	raw

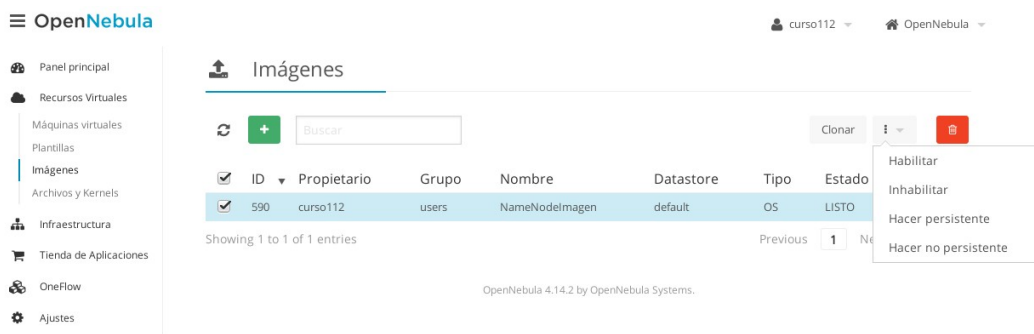
At the bottom of the table, it says 'Showing 1 to 7 of 7 entries'. There are also navigation links: Previous, 1 (selected), Next, and a dropdown menu showing 10.

OpenNebula 4.14.2 by OpenNebula Systems.

Al darle a importar, se mostrará una pantalla para editar diferentes parámetros. En este caso se pondrá como nombre de la imagen NameNodeImagen y, como nombre de plantilla, NameNodePlantilla.

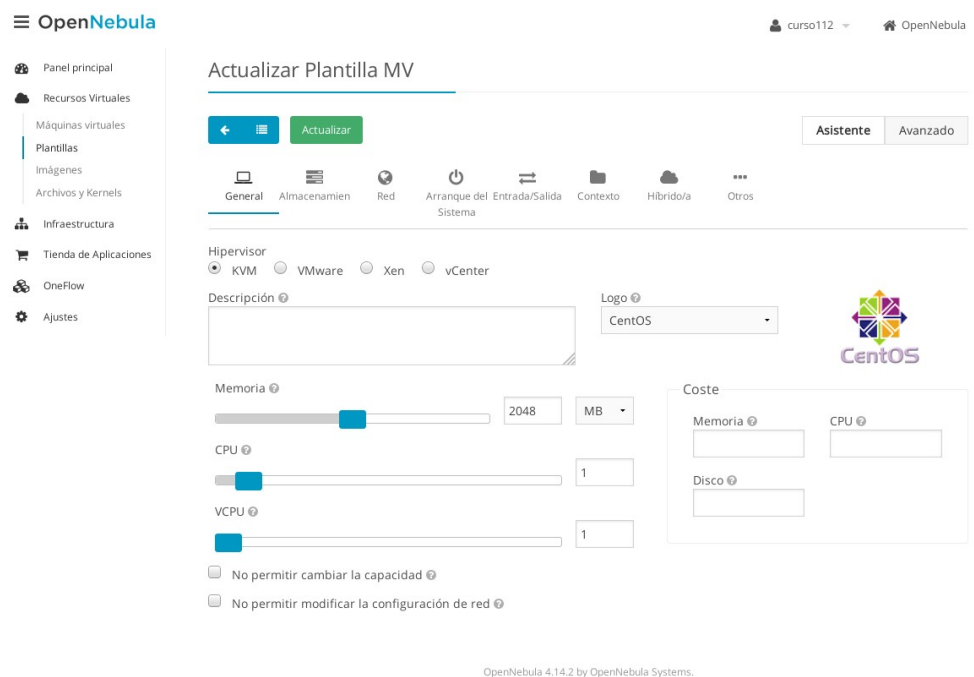
Tras importarla, accedemos a la sección "Imágenes", que se encuentra bajo "Recursos Virtuales". En esta pantalla aparecerá la imagen recién importada en estado "BLOQUEADA" en un primer momento. Será necesario esperar hasta que el estado cambie a "LISTO" y, en ese momento, seleccionaremos la opción "Hacer persistente" del menú desplegable de la derecha.

Para comprobar que este paso se ha realizado correctamente, se puede hacer click en el nombre de la imagen para que aparezca su información.

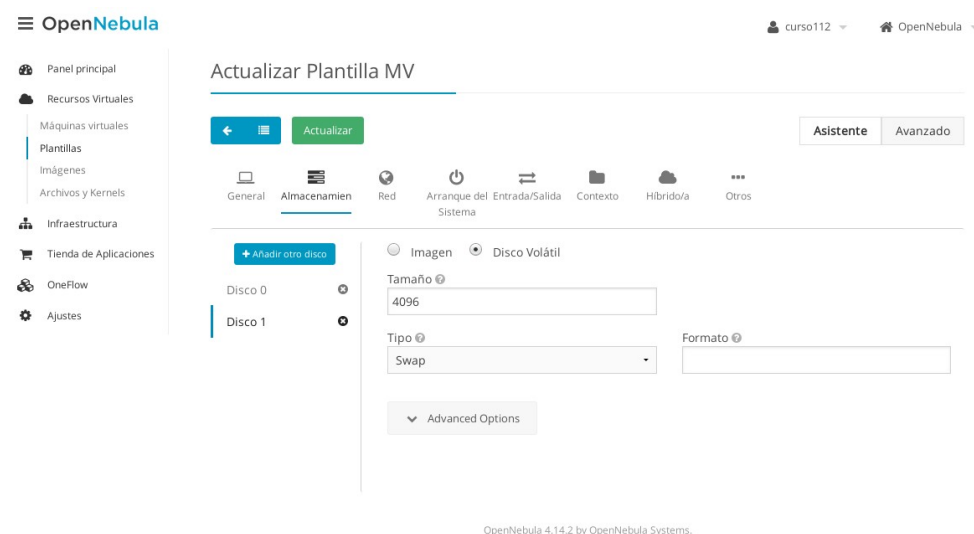


El siguiente paso será acceder a "Plantillas", seleccionar la plantilla creada y darle a actualizar para editarla.

En "General", se modificará la memoria a 2048MB.



En "Almacenamiento" se añadirá un nuevo disco de tipo Disco Volátil con tamaño 4096 y tipo Swap.



En "Contexto" se indicará el contenido de un fichero de clave pública creado previamente para poder realizar la conexión por ssh.

OpenNebula

curso112 OpenNebula

Actualizar Plantilla MV

Actualizar

Asistente Avanzado

General Almacenamiento Red Arranque del Sistema Entrada/Salida Contexto Híbrido/a Otros

Configuración

Archivos

Variables definidas por el usuario

☒ Añadir contextualización SSH

Clave Pública:

/qkvoI+IHjGYUUXHYVqobLpUvRD3o/
edgar@debian

☒ Añadir contextualización de Red

☒ Añadir token de OneGate

Ejecutar Script

☐ Script Codificado en Base64

Entradas de usuario

Nombre	Tipo	Descripción
+ Agrega otro atributo		

OpenNebula 4.14.2 by OpenNebula Systems.

Por último, será necesario darle al botón "Actualizar" para persistir todos estos cambios.

Una vez editadas la imagen y la plantilla, procederemos a crear la máquina virtual desde la sección "Máquinas Virtuales", dándole al botón con el signo "+". Seleccionaremos la plantilla y le pondremos el nombre de NameNode a la máquina virtual.

Crear Máquina Virtual

Restablecer Crear

Seleccione una plantilla

Buscar

ID	Propietario	Grupo	Nombre	Fecha de registro
462	curso112	users	NameNodePlantilla	18:34:09 04/10/2017

Showing 1 to 1 of 1 entries

Previous 1 Next 10

Ha seleccionado la siguiente Plantilla: NameNodePlantilla

Nombre de MV

NameNode

Número de instancias

1

☐ Mantener

Tras editar el nombre, darle a crear. La máquina tardará unos minutos en arrancar y estará lista cuando su estado pase de PENDING a EJECUTANDO.

Una vez lista, nos podremos conectar a la máquina a través de ssh ejecutando:

```
ssh -i path_al_fichero_id_rsa root@ip_máquina_virtual
```

Donde *path_al_fichero_id_rsa* es la ruta a la clave privada generada junto a la clave pública usada en la configuración de la máquina virtual y *ip_maquina_virtual* es la ip de la máquina recién creada y que podemos ver en desde la sección "Máquinas virtuales".

```
edgar@debian:~/Documentos/master/TCDM/Practicas/P1$ ssh -i ~/.ssh/id_rsa root@10.38.3.55
The authenticity of host '10.38.3.55 (10.38.3.55)' can't be established.
ECDSA key fingerprint is 98:f3:61:36:0d:bf:89:66:7d:c5:8a:bd:65:bf:b5:54.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.38.3.55' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

42 packages can be updated.
13 updates are security updates.

Last login: Mon May 23 13:52:18 2016 from 193.144.44.57
root@localhost:~#
```

2. Descarga e instalación de Hadoop

Tras acceder a la máquina, se deberá instalar Java y otras librerías que usa Hadoop. Para ello ejecutamos lo siguiente:

```
apt-get update
```

```

Kali Linux - kali@kali:~$ apt-get update
52 packages can be updated.
13 updates are security updates.

Last login: Mon May 23 13:52:18 2016 from 193.144.44.57
kali@kali:~$ sudo apt-get update
Hit:1 http://archive.ubuntu.com/ubuntu xenial InRelease
Hit:2 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Hit:3 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Hit:4 http://archive.ubuntu.com/ubuntu xenial-backports InRelease [102 kB]
Get:5 http://archive.ubuntu.com/ubuntu xenial-updates/main Sources [277 kB]
Get:6 http://archive.ubuntu.com/ubuntu xenial-updates/restricted Sources [3,404 B]
Get:7 http://archive.ubuntu.com/ubuntu xenial-updates/universe Sources [175 kB]
Get:8 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse Sources [7,203 B]
Get:9 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [636 kB]
Get:10 http://archive.ubuntu.com/ubuntu xenial-updates/main 1386 Packages [606 kB]
Get:11 http://archive.ubuntu.com/ubuntu xenial-updates/main Translation-en [250 kB]
Get:12 http://security.ubuntu.com/ubuntu xenial-security/main Sources [93.1 kB]
Get:13 http://archive.ubuntu.com/ubuntu xenial-updates/restricted amd64 Packages [7,972 B]
Get:14 http://archive.ubuntu.com/ubuntu xenial-updates/restricted 1386 Packages [7,988 B]
Get:15 http://archive.ubuntu.com/ubuntu xenial-updates/restricted Translation-en [2,692 B]
Get:16 http://archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [558 kB]
Get:17 http://security.ubuntu.com/ubuntu xenial-security/restricted Sources [2,680 B]
Get:18 http://archive.ubuntu.com/ubuntu xenial-updates/universe 1386 Packages [515 kB]
Get:19 http://archive.ubuntu.com/ubuntu xenial-updates/universe Translation-en [218 kB]
Get:20 http://security.ubuntu.com/ubuntu xenial-security/universe Sources [43.0 kB]
Get:21 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [15.3 kB]
Get:22 http://security.ubuntu.com/ubuntu xenial-security/multiverse Sources [1,340 B]
Get:23 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse 1386 Packages [14.5 kB]
Get:24 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse Translation-en [7,544 B]
Get:25 http://archive.ubuntu.com/ubuntu xenial-backports/main Sources [3,432 B]
Get:26 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [558 kB]
Get:27 http://archive.ubuntu.com/ubuntu xenial-backports/universe Sources [4,376 B]
Get:28 http://archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [4,860 B]
Get:29 http://archive.ubuntu.com/ubuntu xenial-backports/main 1386 Packages [4,852 B]
Get:30 http://archive.ubuntu.com/ubuntu xenial-backports/main Translation-en [3,220 B]
Get:31 http://archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [5,996 B]
Get:32 http://archive.ubuntu.com/ubuntu xenial-backports/universe 1386 Packages [6,006 B]
Get:33 http://security.ubuntu.com/ubuntu xenial-security/main 1386 Packages [133 kB]
Get:34 http://archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en [3,966 B]
Get:35 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [159 kB]
Get:36 http://security.ubuntu.com/ubuntu xenial-security/restricted amd64 Packages [7,352 B]
Get:37 http://security.ubuntu.com/ubuntu xenial-security/restricted 1386 Packages [7,364 B]
Get:38 http://security.ubuntu.com/ubuntu xenial-security/restricted Translation-en [2,122 B]
Get:39 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [169 kB]
Get:40 http://security.ubuntu.com/ubuntu xenial-security/universe 1386 Packages [148 kB]
Get:41 http://security.ubuntu.com/ubuntu xenial-security/universe Translation-en [89.9 kB]
Get:42 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [2,760 B]
Get:43 http://security.ubuntu.com/ubuntu xenial-security/multiverse 1386 Packages [2,916 B]
Get:44 http://security.ubuntu.com/ubuntu xenial-security/multiverse Translation-en [1,236 B]
Fetched 5,661 kB in 3s (1,499 kB/s)
Reading package lists... Done
root@kali:~$

```

```
apt-get install openjdk-8-jre libssl-dev
```

[illegible]

El siguiente paso será instalar Hadoop. Para ello creamos un directorio y descargamos Hadoop en él, descomprimos el fichero descargado, creamos un enlace simbólico y definimos la variable `HADOOP_PREFIX`:

```
mkdir /opt/bd
cd /opt/bd
wget http://apache.uvigo.es/hadoop/common/stable/hadoop-2.7.4.tar.gz
tar xvzf hadoop-2.7.4.tar.gz
rm hadoop-2.7.4.tar.gz
ln -s hadoop-2.7.4 hadoop
export HADOOP_PREFIX=/opt/bd/hadoop
```

Después, creamos el grupo `hadoop`, un usuario `hdmaster` y cambiamos el propietario del directorio `/opt/bd`:

```
groupadd -r hadoop
useradd -r -g hadoop -d /opt/bd -s /bin/bash hdmaster
chown -R hdmaster:hadoop /opt/bd
```

A continuación, creamos directorios para los datos de HDFS y cambiamos el propietario a `hdmaster`:

```
mkdir -p /var/data/hadoop/hdfs/nn
mkdir -p /var/data/hadoop/hdfs/cpn
mkdir -p /var/data/hadoop/hdfs/dn
chown -R hdmaster:hadoop /var/data/hadoop/hdfs
```

También creamos los directorios para los ficheros de log:

```
mkdir -p /var/log/hadoop/yarn
mkdir -p /var/log/hadoop/hdfs
mkdir -p /var/log/hadoop/mapred
chown -R hdmaster:hadoop /var/log/hadoop
```

Por último, modificamos el fichero `/etc/ssh/ssh_config` poniendo el parámetro `StrictHostKeyChecking` a `no`.

```
root@localhost:~# mkdir /opt/bd
root@localhost:~# ls /opt/
bd
root@localhost:~# cd /opt/bd/
root@localhost:opt/bd# wget http://apache.uvigo.es/hadoop/common/stable/hadoop-2.7.4.tar.gz
--2017-10-04 19:28:51-- http://apache.uvigo.es/hadoop/common/stable/hadoop-2.7.4.tar.gz
Resolving apache.uvigo.es (apache.uvigo.es)... 193.146.32.74, 2001:728:1214:4200::74
Connecting to apache.uvigo.es (apache.uvigo.es)[193.146.32.74]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 266688029 (254M) [application/x-gzip]
Saving to: 'hadoop-2.7.4.tar.gz'

hadoop-2.7.4.tar.gz                               100%[=====] 254.33M  68.4MB/s   in 3.9s
2017-10-04 19:28:55 (65.9 MB/s) - 'hadoop-2.7.4.tar.gz' saved [266688029/266688029]
root@localhost:opt/bd#
```

```
hadoop-2.7.4/etc/hadoop/hadoop-metrics2.properties
hadoop-2.7.4/etc/hadoop/mapred-env.sh
hadoop-2.7.4/etc/hadoop/capacity-scheduler.xml
hadoop-2.7.4/etc/hadoop/https-log4j.properties
root@localhost:opt/bd# ls
hadoop-2.7.4 hadoop-2.7.4.tar.gz
root@localhost:opt/bd# rm hadoop-2.7.4.tar.gz
root@localhost:opt/bd# ln -s hadoop-2.7.4/
bin/      etc/      include/  lib/      libexec/  LICENSE.txt NOTICE.txt README.txt sbin/     share/    src/
root@localhost:opt/bd# ln -s hadoop-2.7.4 hadoop
root@localhost:opt/bd# export HADOOP_PREFIX=/opt/bd/hadoop
root@localhost:opt/bd# groupadd -r hadoop
root@localhost:opt/bd# useradd -r -g hadoop -d /opt/bd/ -s /bin/bash hdmaster
root@localhost:opt/bd# chown -R hdmaster:hadoop /opt/bd/
root@localhost:opt/bd# mkdir -p /var/data/hadoop/hdfs/nn
root@localhost:opt/bd# mkdir -p /var/data/hadoop/hdfs/cpn
root@localhost:opt/bd# mkdir -p /var/data/hadoop/hdfs/dn
root@localhost:opt/bd# chown -R hdmaster:hadoop /var/data/hadoop/hdfs
root@localhost:opt/bd# ls
hadoop hadoop-2.7.4
root@localhost:opt/bd# ls -l
total 4
lrwxrwxrwx 1 hdmaster hadoop 12 Oct 4 19:32 hadoop -> hadoop-2.7.4
drwxr-xr-x 10 hdmaster hadoop 4096 Aug 1 03:09 hadoop-2.7.4
root@localhost:opt/bd#
```



```

hdmaster@localhost:/opt/bd$ vim ~/.bashrc
hdmaster@localhost:/opt/bd$ . ~/.bashrc
hdmaster@localhost:/opt/bd$ hadoop version
Hadoop 2.7.4
Subversion https://shv@git-wip-us.apache.org/repos/asf/hadoop.git -r cd915e1e8d9d0131462a0b7301586c175728a282
Compiled by kshvachk on 2017-08-01T00:29Z
Compiled with protoc 2.5.0
From source with checksum 50b0468318b4ce9bd24dc467b7ce1148
This command was run using /opt/bd/hadoop-2.7.4/share/hadoop/common/hadoop-common-2.7.4.jar
hdmaster@localhost:/opt/bd$

```

Ejecutamos lo siguiente para que el usuario hdmaster se pueda conectar a los nodos del cluster por ssh:

```

ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
chmod 644 ~/.ssh/authorized_keys
ssh localhost

```

```

Bcheiro Editor Ver Terminal Tabs Ayuda
This command was run using /opt/bd/hadoop-2.7.4/share/hadoop/common/hadoop-common-2.7.4.jar
hdmaster@localhost:/opt/bd$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Created directory '/opt/bd/.ssh'.
Your identification has been saved in /opt/bd/.ssh/id_rsa.
Your public key has been saved in /opt/bd/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:JdBFE+YF/12HJZodgNpJ6EhgYFjp3QHR6ae3RyRZg0 hdmaster@localhost
The key's randomart image is:
+---[RSA 2048]-----+
| ..+..E=0+ |
| oo.o=..o* |
| o.+...oBo. |
| .. o + *oX |
| o . =S= + |
| ..+..E=0+ |
+---[SHA256]-----+
hdmaster@localhost:/opt/bd$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hdmaster@localhost:/opt/bd$ chmod 644 ~/.ssh/authorized_keys
hdmaster@localhost:/opt/bd$
edgar@debian: ~/Documentos/master/TCDM/Practicas/P1

```

4. Configuración de los demonios de Hadoop

Como usuario hdmaster, debemos modificar los siguientes ficheros: core-site.xml, hdfs-site.xml, yarn-site.xml, mapred-site.xml.

- core-site.xml:

```

Bcheiro Editor Ver Terminal Tabs Ayuda
<?xml version="1.0" encoding="UTF-8"?>
<xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://namenode:9000/</value>
    <final>true</final>
  </property>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>/var/tmp</value>
    <final>true</final>
  </property>
  <property>
    <name>hadoop.http.staticuser.user</name>
    <value>hdfs</value>
    <final>true</final>
  </property>
</configuration>

-- INSERT --
38,14 All
edgar@debian: ~/Documentos/master/TCDM/vpn

```

- hdfs-site.xml:

```

Echeiro Editor VIM Terminal Tabs Ayuda
<configuration>
<property>
  <!-- Factor de replicación de los bloques -->
  <name>dfs.replication</name>
  <value>2</value>
  <final>true</final>
</property>
<property>
  <!-- Tamaño del bloque (uno defecto 128m) -->
  <name>dfs.blocksize</name>
  <value>64m</value>
  <final>true</final>
</property>
<property>
  <!-- Lista (separada por comas) de directorios donde el namenode guarda los metadatos.
  En un sistema real debería incluir por lo menos dos directorios:
  - /etc/hadoop/conf/: el namenode y otro remoto montado por NFS -->
  <name>dfs.namenode.name.dir</name>
  <value>file:///var/data/hadoop/hdfs/nn</value>
  <final>true</final>
</property>
<property>
  <!-- Lista (separada por comas) de directorios donde el checkpoint node guarda los checkpoints.
  En un sistema real debería incluir por lo menos dos directorios:
  - /etc/hadoop/conf/: el checkpoint node y otro remoto montado por NFS -->
  <name>dfs.checkpoint.dir</name>
  <value>file:///var/data/hadoop/hdfs/cpn</value>
  <final>true</final>
</property>
<property>
  <!-- Lista (separada por comas) de directorios donde el checkpoint node guarda los edits temporales -->
  <name>dfs.checkpoint.edits.dir</name>
  <value>file:///var/data/hadoop/hdfs/cpn</value>
  <final>true</final>
</property>
<property>
  <!-- Lista (separada por comas) de directorios donde los datanodes guardan los datos:
  por conveniencia, si los nodos tienen varios discos es conveniente
  usar uno de los discos locales -->
  <name>dfs.datanode.data.dir</name>
  <value>file:///var/data/hadoop/hdfs/dn</value>
  <final>true</final>
</property>
<property>
  <!-- Dirección a puerto del servicio web del namenode -->
  <name>dfs.namenode.http-address</name>
  <value>namenode:50070</value>
  <final>true</final>
</property>
<property>
  <!-- Dirección a puerto del servicio web del checkpoint node (aka secondary namenode) -->
  <name>dfs.namenode.secondary.http-address</name>
  <value>checkpointnode:50090</value>
</property>

```

- yarn-site.xml. Añadiremos la propiedades que aparecen a continuación al resto que se muestran en la captura:

```

<property>
  <!-- El ResourceManager -->
  <name>yarn.resourcemanager.hostname</name>
  <value>resourcemanager</value>
  <final>true</final>
</property>
<property>
  <!-- Indica a los NodeManagers que tienen que implementar el servicio de
  barajado mapreduce -->
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
  <final>true</final>
</property>

```

```
Ficheiro Editar Ver Terminal Tabs Ayuda
<property>
  <!-- Clase que implementa el servicio de barajado mapreduce -->
  <name>yarn.nodemanager.aux-services.mapreduce_shuffle.class</name>
  <value>org.apache.hadoop.mapred.ShuffleHandler</value>
  <final>true</final>
</property>
<property>
  <!-- Numero de cores del Nodemanager (por defecto: 8) -->
  <name>yarn.nodemanager.resource.cpu-cores</name>
  <value>1</value>
  <final>true</final>
</property>
<property>
  <!-- Memoria fisica (MB) que puede ser reservada por los contenedores (por defecto: 8192) -->
  <!-- Para ser menor que la RAM fisica, para que funcionen otros servicios -->
  <name>yarn.nodemanager.resource.memory-mb</name>
  <value>1536</value>
  <final>true</final>
</property>
<property>
  <!-- Raportio de memoria virtual a memoria fisica (por defecto: 2.1) -->
  <name>yarn.nodemanager.vmem-pmem-ratio</name>
  <value>1</value>
  <final>true</final>
</property>
<property>
  <!-- Numero maximo de cores por container (por defecto: 32) -->
  <name>yarn.scheduler.maximum-allocation-vcores</name>
  <value>1</value>
  <final>true</final>
</property>
<property>
  <!-- Memoria reservada permitida por container al Resource Manager (MBs) -->
  <name>yarn.scheduler.minimum-allocation-mb</name>
  <value>512</value>
  <final>true</final>
</property>
<property>
  <!-- Memoria reservada permitida por container al Resource Manager (MBs) -->
  <name>yarn.scheduler.maximum-allocation-mb</name>
  <value>1536</value>
  <final>true</final>
</property>
<property>
  <!-- Habilita agregacion de logs de el historyserver -->
  <name>yarn.log-aggregation-enable</name>
  <value>true</value>
</property>
</configuration>
82,1 Bot
```

- **mapred-site.xml.** Copiamos el fichero **mapred-site.xml.template** a **mapred-site.xml**:

```
Ficheiro Editar Ver Terminal Tabs Ayuda
distributed under the license is distributed on an "AS IS" BASIS,
without warranties or conditions of any kind, either express or implied.
See the license for the specific language governing permissions and
limitations under the license. See accompanying LICENSE file.
<!-- This site-specific property overrides in this file. -->
<configuration>
  <property>
    <!-- Nombre del framework de MapReduce -->
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
    <final>true</final>
  </property>
  <property>
    <!-- Memoria reservada (MB) por map (por defecto: 1536) -->
    <name>mapreduce.map.memory.mb</name>
    <value>768</value>
    <final>true</final>
  </property>
  <property>
    <!-- Memoria reservada (MB) por reduce (por defecto: 3072) -->
    <name>mapreduce.reduce.memory.mb</name>
    <value>768</value>
    <final>true</final>
  </property>
  <property>
    <!-- Memoria reservada (MB) por los maps (por defecto: -Xmx1024M) -->
    <name>mapreduce.map.java.opts</name>
    <value>-Xmx512M</value>
    <final>true</final>
  </property>
  <property>
    <!-- Memoria reservada (MB) por los reduces (por defecto: -Xmx2560M) -->
    <name>mapreduce.reduce.java.opts</name>
    <value>-Xmx512M</value>
    <final>true</final>
  </property>
  <property>
    <name>mapreduce.jobhistory.address</name>
    <value>jobhistoryserver:19820</value>
    <final>true</final>
  </property>
  <property>
    <name>mapreduce.jobhistory.webapp.address</name>
    <value>jobhistoryserver:19888</value>
    <final>true</final>
  </property>
</configuration>
edgar@debian: ~/Documentos/master/TCDM/vpn | hdmaster@localhost: /root
```

Por último, configuraremos los ficheros ***-env.sh** añadiendo lo siguiente:

- **hadoop-env.sh:**
JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
HADOOP_LOG_DIR=/var/log/hadoop/hdfs

- yarn-env.sh

```
JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
YARN_LOG_DIR=/var/log/hadoop/yarn
```

- mapred-env.sh

```
JAVA_HOME= /usr/lib/jvm/java-8-openjdk-amd64
HADOOP_MAPRED_LOG_DIR= /var/log/hadoop/mapred
```

5. Crear el cluster hadoop

Para crear el cluster hadoop, se clonará la imagen y la plantilla creadas anteriormente. Para clonaras, es necesario apagar la máquina virtual desde la web.

OpenNebula 4.14.2 by OpenNebula Systems.

Una vez apagada, en "Imágenes" seleccionaremos la imagen y la clonaremos 3 veces.

OpenNebula 4.14.2 by OpenNebula Systems.

Les pondremos de nombre DataNode1Imagen, DataNode2Imagen y DataNode3Imagen y esperaremos a que estén en estado LISTO.

En plantillas, realizaremos el mismo proceso, llamándoles DataNode1Plantilla, DataNode2Plantilla y DataNode3Plantilla. También les asignaremos a cada una de ellas la imagen que le corresponda cambiando en Disco 0 en el apartado "Almacenamiento".

Actualizar Plantilla MV

Actualizar

Asistente

Avanzado

General Almacenamiento Red Arranque del Sistema Entrada/Salida Contexto Híbrido/a Otros

+ Añadir otro disco

Disco 0

Disco 1

Imagen Disco Volátil

Buscar

ID	Propietario	Grupo	Nombre	Datastore	Tipo	Estado	#MVS
687	curso112	users	DataNode3Imagen	default	OS	LISTO	0
686	curso112	users	DataNode2Imagen	default	OS	LISTO	0
685	curso112	users	DataNode1Imagen	default	OS	LISTO	0
590	curso112	users	NameNodeImagen	default	OS	LISTO	0

Showing 1 to 4 of 4 entries

Previous

1

Next

10

Ha seleccionado la siguiente imagen: DataNode3Imagen

Advanced Options

OpenNebula 4.14.2 by OpenNebula Systems.

Iniciaremos el cluster con sólo 3 nodos utilizando las plantillas NameNodePlantilla, DataNode1Plantilla y DataNode2Plantilla creando las máquinas virtuales NameNode, DataNode1 y DataNode2.

Máquinas virtuales

Buscar

ID	Propietario	Grupo	Nombre	Estado	Host	IPs
1965	curso112	users	DataNode2	EJECUTANDO	c0515	10.38.3.233
1964	curso112	users	DataNode1	EJECUTANDO	c0513	10.38.3.232
1963	curso112	users	NameNode	EJECUTANDO	c0413	10.38.3.231

Showing 1 to 3 of 3 entries

Previous

1

Next

10

3 TOTAL 3 ACTIVA 0 APAGADA 0 PENDIENTE 0 FALLO

OpenNebula 4.14.2 by OpenNebula Systems.

Una vez levantadas, actualizamos el fichero /etc/hosts de cada máquina para poder acceder a ellas por nombre en vez de IP. Podemos ayudarnos de utilidades como clusterssh, que nos permite conectarnos a las tres máquinas y ejecutar comandos a la vez en cada una de ellas.

```

CLUSTERSSH: root@10.38.3.231
127.0.0.1 localhost
127.0.1.1 UbuntuServer16041ts
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
10.38.3.231 namenode.resourcenanager
10.38.3.232 datanode1.checkpointnode
10.38.3.233 datanode2.jobhistoryserve[]

CLUSTERSSH: root@10.38.3.232
127.0.0.1 localhost
127.0.1.1 UbuntuServer16041ts
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
10.38.3.231 namenode.resourcenanager
10.38.3.232 datanode1.checkpointnode
10.38.3.233 datanode2.jobhistoryserve[]

CLUSTERSSH: root@10.38.3.233
127.0.0.1 localhost
127.0.1.1 UbuntuServer16041ts
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
10.38.3.231 namenode.resourcenanager
10.38.3.232 datanode1.checkpointnode
10.38.3.233 datanode2.jobhistoryserve[]

```

También tendremos que editar el fichero /etc/hosts de nuestro equipo para poder acceder a los servicios web por nombre.

Por último, cambia el nombre de las máquinas con el comando hostname.

<pre>root@localhost:~# hostname localhost root@localhost:~# hostname nanenode root@localhost:~# hostname nanenode root@localhost:~# []</pre>	<pre>root@localhost:~# hostname localhost root@localhost:~# hostname datanode1 root@localhost:~# hostname datanode1 root@localhost:~# []</pre>	<pre>root@localhost:~# hostname localhost root@localhost:~# hostname datanode2 root@localhost:~# hostname datanode2 root@localhost:~# []</pre>
--	--	--

Parte II: Uso básico del cluster

1. Inicio de Hadoop

Para iniciar Hadoop, lo primero que debemos hacer es preparar el HDFS. Para ello, nos conectamos al NameNode y, como usuario hdmaster, iniciamos HDFS ejecutando:

```
root@localhost:~# sudo su - hdmaster
hdmaster@namenode:~$ hdfs namenode -format
```

La salida de la ejecución debería indicar lo siguiente:

```
17/10/18 19:00:47 INFO util.GSet: capacity = 2^15 = 32768 entries
17/10/18 19:00:47 INFO namenode.FSImage: Allocated new BlockPoolId: BP-148548056
3-10.38.3.231-1508346047257
17/10/18 19:00:47 INFO common.Storage: Storage directory /var/data/hadoop/hdfs/n
n has been successfully formatted.
17/10/18 19:00:47 INFO namenode.FSImageFormatProtobuf: Saving image file /var/da
ta/hadoop/hdfs/nn/current/fsimage.ckpt_000000000000000000 using no compression
17/10/18 19:00:47 INFO namenode.FSImageFormatProtobuf: Image file /var/data/hado
op/hdfs/nn/current/fsimage.ckpt_000000000000000000 of size 325 bytes saved in 0
seconds.
17/10/18 19:00:48 INFO namenode.NNStorageRetentionManager: Going to retain 1 ima
ges with txid >= 0
17/10/18 19:00:48 INFO util.ExitUtil: Exiting with status 0
17/10/18 19:00:48 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at namenode/10.38.3.231
*****/
hdmaster@namenode:~$
```

A continuación, iniciaremos los demonios. Hay que tener en cuenta el papel de cada máquina para iniciar los demonios correspondientes.

```
17/10/18 19:00:48 INFO util.ExitUtil: Exiting with status 0
17/10/18 19:00:48 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at namenode/10.38.3.231
*****/
hdmaster@namenode:~$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start namenode
Starting namenode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-namenode-nam
enode.out
hdmaster@namenode:~$ $HADOOP_PREFIX/sbin/yarn-daemon.sh start resourcemanager
Starting resourcemanager, logging to /var/log/hadoop/yarn/yarn-hdmaster-resourc
e-manager-namenode.out
hdmaster@namenode:~$ tail /var/log/hadoop/hdfs/

root@localhost:~# su - hdmaster
hdmaster@datanode1:~$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start datanode
Starting datanode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-datanode-data
node1.out
hdmaster@datanode1:~$ $HADOOP_PREFIX/sbin/yarn-daemon.sh start nodemanager
Starting nodemanager, logging to /var/log/hadoop/yarn/yarn-hdmaster-nodemanager-
datanode1.out
hdmaster@datanode1:~$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start secondarynamen
ode
Starting secondarynamenode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-seco
ndarynamenode-datanode1.out
hdmaster@datanode1:~$

root@localhost:~# su - hdmaster
hdmaster@datanode2:~$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start datanode
Starting datanode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-datanode-data
node2.out
hdmaster@datanode2:~$ $HADOOP_PREFIX/sbin/yarn-daemon.sh start nodemanager
Starting nodemanager, logging to /var/log/hadoop/yarn/yarn-hdmaster-nodemanager-
datanode2.out
hdmaster@datanode2:~$ $HADOOP_PREFIX/sbin/yarn-jobhistory-daemon.sh start historys
erver
Starting historyserver, logging to /var/log/hadoop/hadoop-hdmaster-histor
yserver-datanode2.out
hdmaster@datanode2:~$
```

Podemos consultar los ficheros de log para ver si ha ocurrido algún error.

```
hdmaster@namenode:~$ tail /var/log/hadoop/hdfs/hadoop-hdmaster-namenode-namenode.log
2017-10-18 19:10:56,114 INFO org.apache.hadoop.hdfs.server.namenode.FSEditLog: Number of transactions: 6 Total time for transactions(ms): 70 Number of transactions batched in Syncs: 0 Number of syncs: 2 SyncTimes(ms): 67
2017-10-18 19:11:21,342 INFO org.apache.hadoop.hdfs.server.namenode.FSNamesystem: Roll Edit Log from 10.38.3.232
2017-10-18 19:11:21,343 INFO org.apache.hadoop.hdfs.server.namenode.FSEditLog: Rolling edit logs
2017-10-18 19:11:21,344 INFO org.apache.hadoop.hdfs.server.namenode.FSEditLog: Ending log segment 1
2017-10-18 19:11:21,374 INFO org.apache.hadoop.hdfs.server.namenode.FSEditLog: Number of transactions: 3 Total time for transactions(ms): 77 Number of transactions batched in Syncs: 0 Number of syncs: 6 SyncTimes(ms): 146
2017-10-18 19:11:21,377 INFO org.apache.hadoop.hdfs.server.namenode.FileJournalManager: Finalizing edits File /var/data/hadoop/hdfs/nn/current/edits_inprogress_00000000000000000001 -> /var/data/hadoop/hdfs/nn/current/edits_0
00000000000000000002-00000000000000000003
2017-10-18 19:11:21,387 INFO org.apache.hadoop.hdfs.server.namenode.FSEditLog: Starting log segment at 10
2017-10-18 19:11:25,086 INFO org.apache.hadoop.hdfs.server.namenode.TransferFsImage: Transfer took 0.14s at 0.00 KB/s
2017-10-18 19:11:25,086 INFO org.apache.hadoop.hdfs.server.namenode.TransferFsImage: Downloaded file fsimage.ckpt_00000000000000000009 size 734 bytes.
2017-10-18 19:11:25,139 INFO org.apache.hadoop.hdfs.server.namenode.NNStorageRetentionManager: Going to retain 2 images with txid >= 0
hdmaster@namenode:~$ tail /var/log/hadoop/hdfs/
hadoop-hdmaster-namenode-namenode.log hadoop-hdmaster-namenode-namenode.out SecurityAuth-hdmaster.audit
hdmaster@namenode:~$ tail /var/log/hadoop/
hdfs/ mapred/ yarn/
hdmaster@namenode:~$ tail /var/log/hadoop/yarn/yarn-hdmaster-resourcemanager-namenode.log
2017-10-18 19:09:22,447 INFO org.apache.hadoop.yarn.webapp.WebHops: Web app cluster started at 8088
2017-10-18 19:09:22,488 INFO org.apache.hadoop.yarn.server.resourcemanager.rmnode.RMNodeImpl: datanode1:33252 Node Transitioned from NEW to RUNNING
2017-10-18 19:09:22,489 INFO org.apache.hadoop.yarn.server.resourcemanager.rmnode.RMNodeImpl: datanode2:44498 Node Transitioned from NEW to RUNNING
2017-10-18 19:09:22,500 INFO org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler: Added node datanode1:33252 clusterResource: <memory:1536, vCores:1>
2017-10-18 19:09:22,501 INFO org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler: Added node datanode2:44498 clusterResource: <memory:3072, vCores:2>
2017-10-18 19:09:22,545 INFO org.apache.hadoop.ipc.callQueueManager: Using callQueue: class java.util.concurrent.LinkedBlockingQueue queueCapacity: 100
2017-10-18 19:09:22,547 INFO org.apache.hadoop.ipc.Server: Starting Socket Reader #1 for port 8033
2017-10-18 19:09:22,554 INFO org.apache.hadoop.yarn.factory.impl.pb.RpcServerFactoryPBImpl: Adding protocol org.apache.hadoop.yarn.server.api.ResourceManagerAdministrationProtocolPB to the server
2017-10-18 19:09:22,555 INFO org.apache.hadoop.ipc.Server: IPC Server Responder: starting
2017-10-18 19:09:22,556 INFO org.apache.hadoop.ipc.Server: IPC Server listener on 8033: starting
hdmaster@namenode:~$
```

Una vez iniciados los demonios, deberíamos tener acceso a los diferentes interfaces de HDFS, YARN, CheckPoint node y JobHistory server a través de:

- <http://namenode:50070>

Namenode Information - Mozilla Firefox

namenode:50070/dfshealth.html#tab-overview

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Overview 'namenode:9000' (active)

Started:	Wed Oct 18 19:08:16 CEST 2017
Version:	2.7.4, rcd915e1e8d9d0131462a0b7301586c175728a282
Compiled:	2017-08-01T00:29Z by kshvachk from branch-2.7.4
Cluster ID:	CID-ce6e2ce5-4e48-4f91-8c7d-8aab1522cef7
Block Pool ID:	BP-1485480563-10.38.3.231-1508346047257

Summary

Security is off.
Safemode is off.
7 files and directories, 0 blocks = 7 total filesystem object(s).
Heap Memory used 30.87 MB of 48.84 MB Heap Memory. Max Heap Memory is 966.69 MB.
Non Heap Memory used 44.67 MB of 45.81 MB Committed Non Heap Memory. Max Non Heap Memory is -1 B.

Configured Capacity:	39.12 GB
DFS Used:	56 KB (0%)
Non DFS Used:	6.02 GB
DFS Remaining:	31.07 GB (79.43%)
Block Pool Used:	56 KB (0%)
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	2 (Decommissioned: 0)
Dead Nodes	0 (Decommissioned: 0)

- <http://namenode:8088>

All Applications - Mozilla Firefox

namenode:8088/cluster

hadoop

All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCoers Used	VCoers Total	VCoers Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
0	0	0	0	0	0 B	3 GB	0 B	2	0	0	2	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:512, vCores:1>	<memory:1536, vCores:1>

Show 20 entries

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Progress	Tracking UI	Blacklisted Nodes
No data available in table											

Showing 0 to 0 of 0 entries

- <http://jobhistoryserver:19888/>

JobHistory - Mozilla Firefox

jobhistoryserver:19888/jobhistory

hadoop

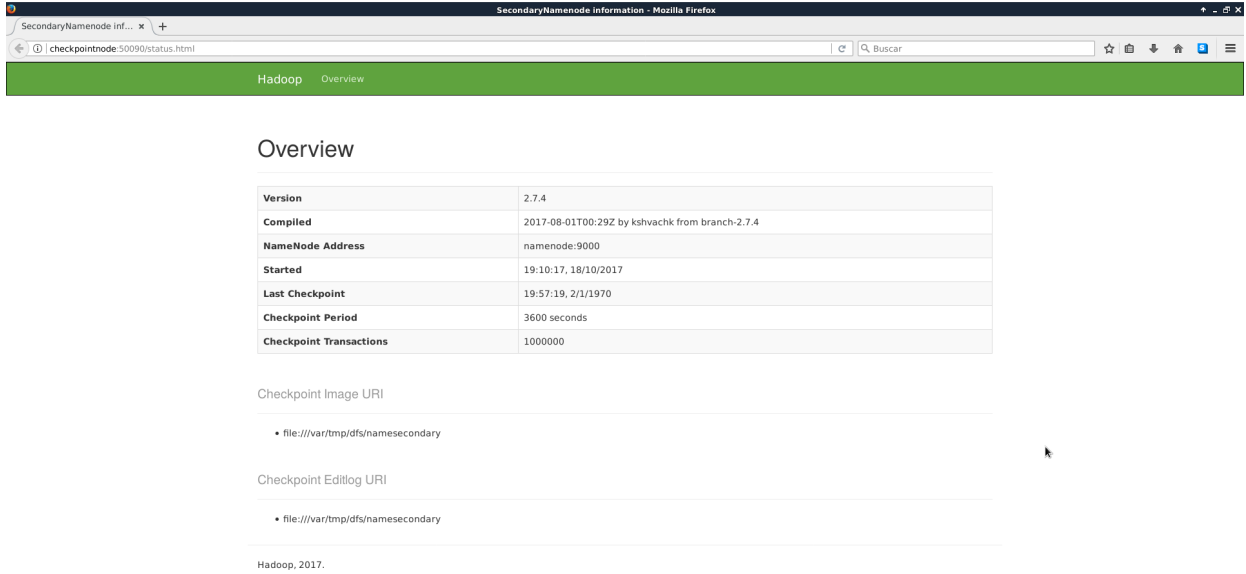
JobHistory

Retired Jobs

Submit Time	Start Time	Finish Time	Job ID	Name	User	Queue	State	Maps Total	Maps Completed	Reduces Total	Reduces Completed
No data available in table											

Showing 0 to 0 of 0 entries

- <http://checkpointnode:50090>



También es posible iniciar los demonios de los DataNodes desde el NameNode. Para poder hacerlo, es necesario editar el fichero `$HADOOP_PREFIX/etc/hadoop/slaves` del NameNode y añadir las IPs de los DataNodes.



Después de editarlo, se pueden iniciar los demonios ejecutando:

```
hdmaster@namenode:~$ vim /opt/bd/hadoop/etc/hadoop/slaves
hdmaster@namenode:~$ $HADOOP_PREFIX/sbin/hadoop-daemons.sh start datanode
10.38.3.233: Warning: Permanently added '10.38.3.233' (ECDSA) to the list of
known hosts.
10.38.3.232: Warning: Permanently added '10.38.3.232' (ECDSA) to the list of
known hosts.
10.38.3.232: datanode running as process 2295. Stop it first.
10.38.3.233: datanode running as process 2632. Stop it first.
hdmaster@namenode:~$ $HADOOP_PREFIX/sbin/yarn-daemons.sh start nodemanager
10.38.3.233: nodemanager running as process 2725. Stop it first.
10.38.3.232: nodemanager running as process 2387. Stop it first.
hdmaster@namenode:~$
```


En el caso de que fuese necesario parar los demonios, se ejecutarían los mismos comandos que para iniciarlos, tan sólo habría que cambiar la palabra start por stop.

Para probar que todo se ha iniciado correctamente, se puede ejecutar un ejemplo de MapReduce haciendo lo siguiente desde el NameNode:

```
hdmaster@namenode:~$ export YARN_EXAMPLES=$HADOOP_PREFIX/share/hadoop/mapredu
ce
hdmaster@namenode:~$ yarn jar $YARN_EXAMPLES/hadoop-mapreduce-examples-2.7.*.j
ar pi 4 10
```

Se puede comprobar su progreso tanto desde la consola, como desde la interfaz web de YARN.

```
CLUSTERSSH: root@10.38.3.231
ar pi 4 10
Number of Maps = 4
Samples per Map = 10
Wrote input for Map #0
Wrote input for Map #1
Wrote input for Map #2
Wrote input for Map #3
Starting Job
17/10/18 19:48:55 INFO client.RMProxy: Connecting to ResourceManager at resour
cemanager/10.38.3.231:8032
17/10/18 19:48:57 INFO input.FileInputFormat: Total input paths to process : 4
17/10/18 19:48:57 INFO mapreduce.JobSubmitter: number of splits:4
17/10/18 19:48:57 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_
1508346555226_0001
17/10/18 19:48:58 INFO impl.YarnClientImpl: Submitted application application_
1508346555226_0001
17/10/18 19:48:59 INFO mapreduce.Job: The url to track the job: http://resourc
emanager:8088/proxy/application_1508346555226_0001/
17/10/18 19:48:59 INFO mapreduce.Job: Running job: job_1508346555226_0001
17/10/18 19:49:19 INFO mapreduce.Job: Job job_1508346555226_0001 running in ub
er mode : false
17/10/18 19:49:19 INFO mapreduce.Job: map 0% reduce 0%
17/10/18 19:49:36 INFO mapreduce.Job: map 25% reduce 0%
```



hadoop

All Applications

Logged in as: hdls

Cluster

About Nodes Node Labels Applications

NEW NEW SAVING SUBMITTED ACCEPTED RUNNING FINISHED FAILED KILLED Scheduler

Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
1	0	1	0	2	2.50 GB	3 GB	0 B	2	2	0	2	0	0	0	0

Scheduler Metrics


Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:512, vCores:1>	<memory:1536, vCores:1>

Show 20 entries

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Progress	Tracking UI	Blacklisted Nodes
application_1508346555226_0001	hdmaster	QuasiMonteCarlo	MAPREDUCE	default	Wed Oct 18 19:48:58 +0200 2017	N/A	RUNNING	UNDEFINED	<div></div>	ApplicationMaster	0

Showing 1 to 1 of 1 entries

Así como comprobar la información de ejecución en el JobHistory server una vez terminado.



hadoop

JobHistory

Logged in as: hdls

Application

About Jobs

Tools

Retired Jobs

Show 20 entries

Search:

Submit Time	Start Time	Finish Time	Job ID	Name	User	Queue	State	Maps Total	Maps Completed	Reduces Total	Reduces Completed
2017.10.18 19:48:58 CEST	2017.10.18 19:49:18 CEST	2017.10.18 19:50:11 CEST	job_1508346555226_0001	QuasiMonteCarlo	hdmaster	default	SUCCEEDED	4	4	1	1

Showing 1 to 1 of 1 entries

2. Uso como usuario no privilegiado

En primer lugar crearemos un nuevo usuario en el NameNode con adduser.

```
root@namenode:~# adduser edgar
Adding user `edgar' ...
Adding new group `edgar' (1001) ...
Adding new user `edgar' (1001) with group `edgar' ...
Creating home directory `/home/edgar' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for edgar
Enter the new value, or press ENTER for the default
  Full Name []: Edgar
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n]
root@namenode:~#
```

Como usuario hdmaster, crearemos los directorios en HDFS del nuevo usuario, les asignaremos el propietario y daremos permisos en /tmp para que el usuario pueda lanzar aplicaciones MapReduce.

```
root@namenode:~# su - hdmaster
hdmaster@namenode:~$ hdfs dfs -mkdir -p /user/edgar
hdmaster@namenode:~$ hdfs dfs -chown edgar /user/edgar
hdmaster@namenode:~$ hdfs dfs -ls /user/edgar
hdmaster@namenode:~$ hdfs dfs -chmod -R 1777 /tmp
hdmaster@namenode:~$
```

A continuación se puede realizar una prueba para comprobar el acceso a HDFS por parte del nuevo usuario. Será necesario ejecutar lo siguiente:

```
hdmaster@namenode:~$ exit
logout
root@namenode:~# su - edgar
edgar@namenode:~$ export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
edgar@namenode:~$ export HADOOP_PREFIX=/opt/bd/hadoop
edgar@namenode:~$ export PATH=$PATH:$HADOOP_PREFIX/bin
edgar@namenode:~$ . ~/.bashrc
edgar@namenode:~$
```

Tras lo cual podemos ejecutar `hdfs dfs -ls` para comprobar el acceso.

```
edgar@namenode:~$ hdfs dfs -ls
edgar@namenode:~$
```

Este comando no devuelve nada porque el usuario no tiene ningún fichero en HDFS. Por lo tanto, el siguiente paso es añadir ficheros.

Ejecutamos `wget` <https://tinyurl.com/TCDM-libros>.

```
s07/wA&slrid=e638249e-30fb-4000-7c39-49e2d5c4a01d [following]
--2017-10-18 23:26:41-- https://nubeusc-my.sharepoint.com/personal/tf_pena_usc_es/_layouts/15/download.aspx?docid=1786d96d35b064aadb0c7276a2aab98db&authkey=Aff1-jPUhh9Wk5JrQks07wA&slrid=e638249e-30fb-4000-7c39-49e2d5c4a01d
Reusing existing connection to nubeusc-my.sharepoint.com:443.
HTTP request sent, awaiting response... 200 OK
Length: 335585280 (320M) [application/x-tar]
Saving to: 'TCDM-libros'

TCDM-libros          98%[=====>] 316.51M   304KB/s   eta 6s
```

Descomprimos el fichero descargado.

```
CLUSTERSSH: root@10.38.3.231
edgar@namenode:~$ ls
TCDM-libros
edgar@namenode:~$ tar -xvf TCDM-libros
libros/
libros/pg17013.txt.gz
libros/pg16625.txt.gz
libros/pg25807.txt.gz
libros/pg32315.txt.gz
libros/random_words.txt.bz2
libros/pg1619.txt.gz
libros/pg7109.txt.gz
libros/pg2000.txt.gz
libros/pg14329.txt.gz
libros/pg24536.txt.gz
libros/pg25640.txt.gz
libros/pg8870.txt.gz
libros/pg18005.txt.gz
libros/pg5201.txt.gz
libros/pg9980.txt.gz
libros/pg17073.txt.gz
edgar@namenode:~$
```

Y copiamos los datos al HDFS.

```
CLUSTERSSH: root@10.38.3.231
libros/pg9980.txt.gz
libros/pg17073.txt.gz
edgar@namenode:~$ hdfs dfs -put libros .
edgar@namenode:~$ hdfs dfs -ls libros
Found 16 items
-rw-r--r--  2 edgar supergroup  441804 2017-10-18 23:41 libros/pg14329.txt.g
z
-rw-r--r--  2 edgar supergroup  264123 2017-10-18 23:41 libros/pg1619.txt.gz
-rw-r--r--  2 edgar supergroup  455129 2017-10-18 23:41 libros/pg16625.txt.g
z
-rw-r--r--  2 edgar supergroup  939502 2017-10-18 23:41 libros/pg17013.txt.g
z
-rw-r--r--  2 edgar supergroup  737367 2017-10-18 23:41 libros/pg17073.txt.g
z
-rw-r--r--  2 edgar supergroup  219304 2017-10-18 23:41 libros/pg18005.txt.g
z
-rw-r--r--  2 edgar supergroup  813698 2017-10-18 23:41 libros/pg2000.txt.gz
-rw-r--r--  2 edgar supergroup  328494 2017-10-18 23:41 libros/pg24536.txt.g
z
-rw-r--r--  2 edgar supergroup  504188 2017-10-18 23:41 libros/pg25640.txt.g
z
-rw-r--r--  2 edgar supergroup   38194 2017-10-18 23:41 libros/pg25807.txt.g
z
-rw-r--r--  2 edgar supergroup  103986 2017-10-18 23:41 libros/pg32315.txt.g
z
-rw-r--r--  2 edgar supergroup  125693 2017-10-18 23:41 libros/pg5201.txt.gz
-rw-r--r--  2 edgar supergroup   82099 2017-10-18 23:41 libros/pg7109.txt.gz
-rw-r--r--  2 edgar supergroup   99685 2017-10-18 23:41 libros/pg8870.txt.gz
-rw-r--r--  2 edgar supergroup   85187 2017-10-18 23:41 libros/pg9980.txt.gz
-rw-r--r--  2 edgar supergroup 330326458 2017-10-18 23:41 libros/random_words.
txt.bz2
edgar@namenode:~$
```


Comprobamos en la interfaz web (namenode:50070), "Utilities > Browse the filesystem" que cada fichero tiene 2 réplicas y los bloques en los que se ha dividido el fichero grande.

Browse Directory

/user/edgar/libros

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	edgar	supergroup	431.45 KB	23:41:46, 18/10/2017	2	64 MB	pg14329.txt.gz
-rw-r--r--	edgar	supergroup	257.93 KB	23:41:46, 18/10/2017	2	64 MB	pg1619.txt.gz
-rw-r--r--	edgar	supergroup	444.46 KB	23:41:46, 18/10/2017	2	64 MB	pg16625.txt.gz
-rw-r--r--	edgar	supergroup	917.48 KB	23:41:46, 18/10/2017	2	64 MB	pg17013.txt.gz
-rw-r--r--	edgar	supergroup	720.08 KB	23:41:46, 18/10/2017	2	64 MB	pg17073.txt.gz
-rw-r--r--	edgar	supergroup	214.16 KB	23:41:47, 18/10/2017	2	64 MB	pg18005.txt.gz
-rw-r--r--	edgar	supergroup	794.63 KB	23:41:47, 18/10/2017	2	64 MB	pg2000.txt.gz
-rw-r--r--	edgar	supergroup	320.79 KB	23:41:47, 18/10/2017	2	64 MB	pg24536.txt.gz
-rw-r--r--	edgar	supergroup	492.37 KB	23:41:47, 18/10/2017	2	64 MB	pg25640.txt.gz
-rw-r--r--	edgar	supergroup	37.3 KB	23:41:47, 18/10/2017	2	64 MB	pg25807.txt.gz
-rw-r--r--	edgar	supergroup	101.55 KB	23:41:47, 18/10/2017	2	64 MB	pg32315.txt.gz
-rw-r--r--	edgar	supergroup	122.75 KB	23:41:47, 18/10/2017	2	64 MB	pg5201.txt.gz
-rw-r--r--	edgar	supergroup	80.17 KB	23:41:48, 18/10/2017	2	64 MB	pg7109.txt.gz
-rw-r--r--	edgar	supergroup	97.35 KB	23:41:48, 18/10/2017	2	64 MB	pg8870.txt.gz
-rw-r--r--	edgar	supergroup	83.19 KB	23:41:48, 18/10/2017	2	64 MB	pg9980.txt.gz
-rw-r--r--	edgar	supergroup	315.02 MB	23:41:55, 18/10/2017	2	64 MB	random_words.txt.bz2

Hadoop, 2017.

Browse Directory

/user/edgar/libros

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	edgar	supergroup	431.45 KB	23:41:46, 18/10/2017	2	64 MB	pg14329.txt.gz
-rw-r--r--	edgar	supergroup	257.93 KB	23:41:46, 18/10/2017	2	64 MB	pg1619.txt.gz
-rw-r--r--	edgar	supergroup	444.46 KB	23:41:46, 18/10/2017	2	64 MB	pg16625.txt.gz
-rw-r--r--	edgar	supergroup	917.48 KB	23:41:46, 18/10/2017	2	64 MB	pg17013.txt.gz
-rw-r--r--	edgar	supergroup	720.08 KB	23:41:46, 18/10/2017	2	64 MB	pg17073.txt.gz
-rw-r--r--	edgar	supergroup	214.16 KB	23:41:47, 18/10/2017	2	64 MB	pg18005.txt.gz
-rw-r--r--	edgar	supergroup	794.63 KB	23:41:47, 18/10/2017	2	64 MB	pg2000.txt.gz
-rw-r--r--	edgar	supergroup	320.79 KB	23:41:47, 18/10/2017	2	64 MB	pg24536.txt.gz
-rw-r--r--	edgar	supergroup	492.37 KB	23:41:47, 18/10/2017	2	64 MB	pg25640.txt.gz
-rw-r--r--	edgar	supergroup	37.3 KB	23:41:47, 18/10/2017	2	64 MB	pg25807.txt.gz
-rw-r--r--	edgar	supergroup	101.55 KB	23:41:47, 18/10/2017	2	64 MB	pg32315.txt.gz
-rw-r--r--	edgar	supergroup	122.75 KB	23:41:47, 18/10/2017	2	64 MB	pg5201.txt.gz
-rw-r--r--	edgar	supergroup	80.17 KB	23:41:48, 18/10/2017	2	64 MB	pg7109.txt.gz
-rw-r--r--	edgar	supergroup	97.35 KB	23:41:48, 18/10/2017	2	64 MB	pg8870.txt.gz
-rw-r--r--	edgar	supergroup	83.19 KB	23:41:48, 18/10/2017	2	64 MB	pg9980.txt.gz
-rw-r--r--	edgar	supergroup	315.02 MB	23:41:55, 18/10/2017	2	64 MB	random_words.txt.bz2

Hadoop, 2017.

File information - random_words.txt.bz2

Download

Block information -- Block 0

Block ID: 10737418! Block 1

Block Pool ID: BP-14 Block 2 38.3.231-1508346047257

Generation Stamp: Block 3

Size: 67108864 Block 4

Availability:

- datanode2
- datanode1

Close

[illegible]

Añadimos dos nuevas propiedades al fichero hdfs-site.xml con las rutas a los ficheros dfs.include y dfs.exclude.

```
CLUSTERSSH: root@10.38.3.231
<name>dfs.namenode.http-address</name>
<value>namenode:50070</value>
<final>true</final>
</property>
<property>
<!-- Dirección y puerto del interfaz web del checkpoint node (aka secondary
namenode) -->
<name>dfs.namenode.secondary.http-address</name>
<value>checkpointnode:50090</value>
<final>true</final>
</property>
<property>
<name>dfs.hosts</name>
<value>/opt/bd/hadoop/etc/hadoop/dfs.include</value>
<final>true</final>
</property>
<property>
<name>dfs.hosts.exclude</name>
<value>/opt/bd/hadoop/etc/hadoop/dfs.exclude</value>
<final>true</final>
</property>
</configuration>
83,1 Bot
```

Seguimos un proceso análogo para el fichero yarn-site.xml.

```
CLUSTERSSH: root@10.38.3.231
<!-- Solicitudes mayores lanzan una InvalidResourceRequestException.(por def
ecto: 8192) -->
<name>yarn.scheduler.maximum-allocation-mb</name>
<value>1536</value>
<final>true</final>
</property>
<property>
<!-- Permite agregacion de logs en el historyserver -->
<name>yarn.log-aggregation-enable</name>
<value>true</value>
</property>
<property>
<name>yarn.resourcemanager.nodes.include-path</name>
<value>/opt/bd/hadoop/etc/hadoop/yarn.include</value>
<final>true</final>
</property>
<property>
<name>yarn.resourcemanager.nodes.exclude-path</name>
<value>/opt/bd/hadoop/etc/hadoop/yarn.exclude</value>
<final>true</final>
</property>
</configuration>
91,1 Bot
```

Reiniciamos los demonios.

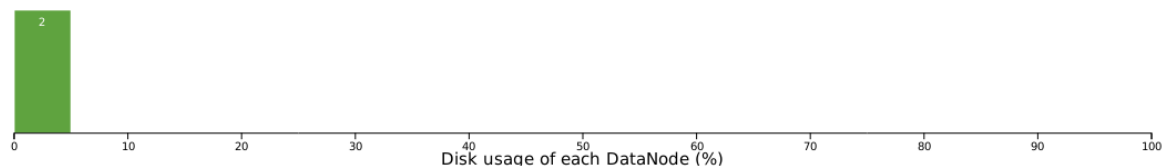
```
hdmaster@namenode:~$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start namenode
starting namenode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-namenode-name
node.out
hdmaster@namenode:~$ $HADOOP_PREFIX/sbin/yarn-daemon.sh start resourcemanager
starting resourcemanager, logging to /var/log/hadoop/yarn/yarn-hdmaster-resource
manager-namenode.out
hdmaster@namenode:~$
```

Empezaremos añadiendo el nombre del nuevo nodo a los ficheros include.

[illegible]

```
hdmaster@namenode:~$ hdfs dfsadmin -refreshNodes
Refresh nodes successful
hdmaster@namenode:~$ yarn rmadmin -refreshNodes
17/10/25 19:05:50 INFO client.RMProxy: Connecting to ResourceManager at resource
manager/10.38.3.231:8033
hdmaster@namenode:~$
```

Datanode Information



Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
datanode1:50010 (10.38.3.232:50010)	0	In Service	19.56 GB	322.93 MB	3.01 GB	15.22 GB	24	322.93 MB (1.61%)	0	2.7.4
datanode2:50010 (10.38.3.233:50010)	0	In Service	19.56 GB	322.93 MB	3.01 GB	15.22 GB	24	322.93 MB (1.61%)	0	2.7.4
datanode3:50010 (10.38.3.67:50010)	Tue Oct 17 2017 00:11:56 GMT+0200 (CEST)	Dead	-	-	-	-	-	-	-	-

Iniciamos el nuevo nodo, modificamos los archivos /etc/hosts de todos los nodos para incluirlo y, desde el nuevo nodo, iniciamos los demonios.

```
root@localhost:~# su - hdmaster
hdmaster@datanode3:~$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start datanode
starting datanode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-datanode-data
node3.out
hdmaster@datanode3:~$ $HADOOP_PREFIX/sbin/yarn-daemon.sh start nodemanager
starting nodemanager, logging to /var/log/hadoop/yarn/yarn-hdmaster-nodemanager-
datanode3.out
hdmaster@datanode3:~$
```

De esta manera, el nodo contacta automáticamente con el NameNode y se une al cluster. Podemos comprobar que el nuevo nodo se ha añadido a través de la línea de comandos o la interfaz web.

```
hdmaster@namenode:~$ hdfs dfsadmin -report
Configured Capacity: 63004594176 (58.68 GB)
Present Capacity: 50038284288 (46.60 GB)
DFS Remaining: 49361027072 (45.97 GB)
DFS Used: 677257216 (645.88 MB)
DFS Used%: 1.35%
Under replicated blocks: 0
Blocks with corrupt replicas: 0
Missing blocks: 0
Missing blocks (with replication factor 1): 0
```

Live datanodes (3):

Name: 10.38.3.233:50010 (datanode2)
Hostname: datanode2
Decommission Status : Normal
Configured Capacity: 21001531392 (19.56 GB)
DFS Used: 338616320 (322.93 MB)
Non DFS Used: 3231846400 (3.01 GB)
DFS Remaining: 16340656128 (15.22 GB)
DFS Used%: 1.61%
DFS Remaining%: 77.81%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Wed Oct 25 19:29:31 CEST 2017

Name: 10.38.3.67:50010 (datanode3)
Hostname: datanode3
Decommission Status : Normal
Configured Capacity: 21001531392 (19.56 GB)
DFS Used: 24576 (24 KB)
Non DFS Used: 3230470144 (3.01 GB)
DFS Remaining: 16680624128 (15.54 GB)
DFS Used%: 0.00%
DFS Remaining%: 79.43%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Wed Oct 25 19:29:30 CEST 2017

Name: 10.38.3.232:50010 (datanode1)
Hostname: datanode1
Decommission Status : Normal
Configured Capacity: 21001531392 (19.56 GB)
DFS Used: 338616320 (322.93 MB)
Non DFS Used: 3232755712 (3.01 GB)
DFS Remaining: 16339746816 (15.22 GB)
DFS Used%: 1.61%
DFS Remaining%: 77.80%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Wed Oct 25 19:29:31 CEST 2017

```
hdmaster@namenode:~$ yarn node -list
17/10/25 19:30:38 INFO client.RMProxy: Connecting to ResourceManager at resourcemanager/10.3
8.3.231:8032
Total Nodes:3
      Node-Id          Node-State Node-Http-Address      Number-of-Running-Containers
datanode2:44498        RUNNING    datanode2:8042         0
datanode1:33252        RUNNING    datanode1:8042         0
datanode3:35553        RUNNING    datanode3:8042         0
hdmaster@namenode:~$
```

El nuevo nodo estará vacío inicialmente, con lo que el cluster estará desbalanceado. Para forzar el balanceo ejecutamos:

```

hdmaster@namenode:~$ hdfs balancer
17/10/29 02:49:45 INFO balancer.Balancer: namenodes = [hdfs://namenode:9000]
17/10/29 02:49:45 INFO balancer.Balancer: parameters = Balancer.Parameters [BalancingPolicy.Node, threshold = 10.0, max idle iteration = 5, number of nodes to be excluded = 0, number of nodes to be included = 0, run during upgrade = false]
Time Stamp          Iteration#  Bytes Already Moved  Bytes Left To Move  Bytes Being Moved
17/10/29 02:49:48 INFO balancer.Balancer: dfs.balancer.movedWinWidth = 5400000 (default=5400000)
17/10/29 02:49:48 INFO balancer.Balancer: dfs.balancer.moverThreads = 1000 (default=1000)
17/10/29 02:49:48 INFO balancer.Balancer: dfs.balancer.dispatcherThreads = 200 (default=200)
17/10/29 02:49:48 INFO balancer.Balancer: dfs.datanode.balance.max.concurrent.moves = 5 (default=5)
17/10/29 02:49:48 INFO balancer.Balancer: dfs.balancer.getBlockSize = 2147483648 (default=2147483648)
17/10/29 02:49:48 INFO balancer.Balancer: dfs.balancer.getBlockSize.min-block-size = 10485760 (default=10485760)
17/10/29 02:49:48 INFO balancer.Balancer: dfs.balancer.max-size-to-move = 10737418240 (default=10737418240)
17/10/29 02:49:49 INFO net.NetworkTopology: Adding a new node: /default-rack/10.38.3.233:50010
17/10/29 02:49:49 INFO net.NetworkTopology: Adding a new node: /default-rack/10.38.3.67:50010
17/10/29 02:49:49 INFO net.NetworkTopology: Adding a new node: /default-rack/10.38.3.232:50010
17/10/29 02:49:49 INFO balancer.Balancer: 0 over-utilized: []
17/10/29 02:49:49 INFO balancer.Balancer: 0 underutilized: []
The cluster is balanced. Exiting...
Oct 29, 2017 2:49:49 AM          0          0 B          0 B
-1 B
Oct 29, 2017 2:49:49 AM Balancing took 4.303 seconds
hdmaster@namenode:~$

```

Para retirar un Datanode debemos añadirlo a los ficheros exclude y ejecutar refreshNodes.

[illegible]


```

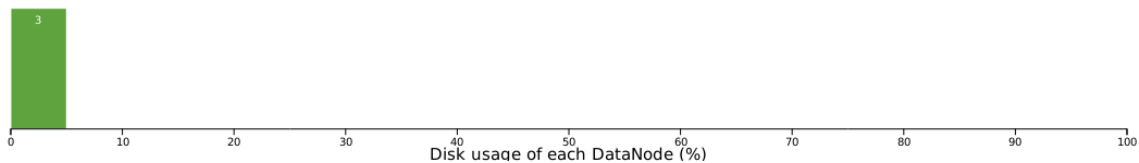
hdmaster@namenode:/opt/bd/hadoop/etc/hadoop$ vim dfs.exclude
hdmaster@namenode:/opt/bd/hadoop/etc/hadoop$ vim yarn.exclude
hdmaster@namenode:/opt/bd/hadoop/etc/hadoop$ hdfs dfsadmin -refreshNodes
Refresh nodes successful
hdmaster@namenode:/opt/bd/hadoop/etc/hadoop$ yarn rmadmin -refreshNodes
17/10/29 02:54:09 INFO client.RMProxy: Connecting to ResourceManager at resourcemanager/10.38.3.231:8033
hdmaster@namenode:/opt/bd/hadoop/etc/hadoop$ █

```

Podemos ver, en este caso desde la interfaz web, que el nodo aparece decomisionado.

Datanode Information

Datanode usage histogram



In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
datanode3:50010 (10.38.3.67:50010)	0	Decommissioned	19.56 GB	28 KB	3.05 GB	15.5 GB	0	28 KB (0%)	0	2.7.4
datanode1:50010 (10.38.3.232:50010)	2	In Service	19.56 GB	322.75 MB	3.05 GB	15.18 GB	22	322.75 MB (1.61%)	0	2.7.4
datanode2:50010 (10.38.3.233:50010)	2	In Service	19.56 GB	322.75 MB	3.05 GB	15.18 GB	22	322.75 MB (1.61%)	0	2.7.4

2. Rack awareness

En este punto configuraremos Hadoop para que conozca la topología de nuestra red y así maximizar el rendimiento. Por defecto, considera que todos los DataNodes están en un mismo rack que identifica como /default-rack. Sin embargo, en clusters multirack debemos indicarle a Hadoop dónde se encuentra cada nodo. En nuestro caso, vamos a suponer que tenemos 2 racks (/rack1 y /rack2).

Para realizar la configuración empezaremos por apagar los demonios en el NameNode como usuario hdmaster.

```

hdmaster@namenode:/opt/bd$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh stop namenode
stopping namenode
hdmaster@namenode:/opt/bd$ $HADOOP_PREFIX/sbin/yarn-daemon.sh stop resourcemanager
stopping resourcemanager
hdmaster@namenode:/opt/bd$ █

```

Creamos el fichero `$HADOOP_PREFIX/etc/hadoop/topology.data` con la siguiente información:

[illegible]

Creamos un script `$HADOOP_PREFIX/etc/hadoop/topology.script` y le damos permisos de ejecución.

```

X CLUSTERSSH: root@10.38.3.231
# !/bin/bash

HADOOP_CONF=/opt/bd/hadoop/etc/hadoop
while [ $# -gt 0 ] ; do
    nodeArg=$1
    exec< ${HADOOP_CONF}/topology.data
    result=""
    while read line ; do
        ar=( $line )
        if [ "${ar[0]}" = "$nodeArg" ] ; then
            result="${ar[1]}"
        fi
    done
    shift
    if [ -z "$result" ] ; then
        echo -n "/default-rack "
    else
        echo -n "$result "
    fi
done
~
~
~
<7.4/etc/hadoop/topology.script" [New] 20L, 341C written 20,4 All

```

```
hdmaster@namenode:/opt/bd$ vim /opt/bd/hadoop/etc/hadoop/topology.script
hdmaster@namenode:/opt/bd$ chmod +x /opt/bd/hadoop/etc/hadoop/topology.script
hdmaster@namenode:/opt/bd$
```

Definimos una nueva propiedad en el fichero core-site.xml y le damos como valor la ruta al script.

```
CLUSTERSSH: root@10.38.3.231
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
  <!-- nombre del Namenode -->
  <name>fs.defaultFS</name>
  <value>hdfs://namenode:9000/</value>
  <final>true</final>
</property>
<property>
  <!-- Almacenamiento temporal (debe tener suficiente espacio) -->
  <name>hadoop.tmp.dir</name>
  <value>/var/tmp</value>
  <final>true</final>
</property>
<property>
  <!-- Usuario por defecto para el interfaz web -->
  <name>hadoop.http.staticuser.user</name>
  <value>hdfs</value>
  <final>true</final>
</property>
<property>
  <name>net.topology.script.file.name</name>
  <value>/opt/bd/hadoop/etc/hadoop/topology.script</value>
  <final>true</final>
</property>
</configuration>
"hadoop-2.7.4/etc/hadoop/core-site.xml" 43L, 1444C written      43,13      Bot
```

Iniciamos los demonios y comprobamos la topología de nuestro cluster.

```
hdmaster@namenode:/opt/bd$ $HADOOP_PREFIX/sbin/hadoop-daemon.sh start namenode
starting namenode, logging to /var/log/hadoop/hdfs/hadoop-hdmaster-namenode-namenode.out
hdmaster@namenode:/opt/bd$ $HADOOP_PREFIX/sbin/yarn-daemon.sh start resourcemanager
starting resourcemanager, logging to /var/log/hadoop/yarn/yarn-hdmaster-resourcemanager-namenode.out
hdmaster@namenode:/opt/bd$ hdfs dfsadmin -printTopology
Rack: /rack1
10.38.3.232:50010 (datanode1)

Rack: /rack2
10.38.3.233:50010 (datanode2)
10.38.3.67:50010 (datanode3)

hdmaster@namenode:/opt/bd$
```